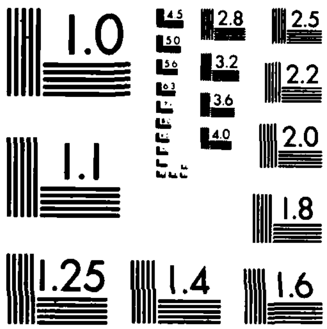
AD-A132 217   A USER'S GUIDE TO ARSPIQ (AUTOREGRESSIVE SPECTRAL   1/2
INFORMATION QUANTILE IDENTIFICATION)(U) TEXAS A AND M
UNIV COLLEGE STATION DEPT OF STATISTICS T J WOODFIELD
UNCLASSIFIED   AUG 83 A-25 DAAG29-83-K-0051   F/G 9/2   NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

TEXAS A&M UNIVERSITY

COLLEGE STATION, TEXAS 77843-3143

STATISTICS
Phone 409 - 845-3141

ADA 132217

A USER'S GUIDE TO ARSPIQ: THE UNIVARIATE TIME SERIES ANALYSIS PROGRAM FOR AUTOREGRESSIVE SPECTRAL INFORMATION QUANTILE IDENTIFICATION

by Terry J. Woodfield

Department of Statistics

Texas A&M University

Technical Report No. A-25

August 1983

Texas A&M Research Foundation
Project No. 4858

"Functional Statistical Data Analysis and Modeling"

Sponsored by the Army Research Office and

the Office of Naval Research

Contract DAAG29-83-K-0051, ONR N00014-83-WRM3008

Professor Emanuel Parzen, Principal Investigator

83 09 06 018

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER A-25	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A User's Guide to ARSPIQ: The Univariate Time Series Analysis Program for Autoregressive Spectral Information Quantile Identification		5. TYPE OF REPORT & PERIOD COVERED Technical 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Terry J. Woodfield		8. CONTRACT OR GRANT NUMBER(s) DAAG29-83-K-0051 ONR N00014-83-WRM3008
9. PERFORMING ORGANIZATION NAME AND ADDRESS Texas A&M University Institute of Statistics College Station, TX 77843		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office P. O. Box 12211 Research Triangle Park, NC 27709		12. REPORT DATE August 1983 13. NUMBER OF PAGES 109
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

NA

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Time series analysis, quantile data analysis, quantile spectral analysis, ARMA representation, model identification, long memory time series, Wolfer's sunspot annual series.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The ARSPIQ (Autoregressive SPectral Information Quantile identification) program is a modified version of ARSPID, a univariate time series program in the TIMESBOARD Computing Library (Newton, 1983). ARSPIQ provides various diagnostics in the time and frequency domains to help one determine whether a time series is long, short, or no memory.

ARSPIQ is written in FORTRAN and consists of a main program and 58 subprograms, many of which are contained in the TIMESBOARD

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-LF-014-6601

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Computing Library. Many of the subprograms used in the quantile analysis are modified versions of those used by the ONESAM program (Parzen and Anderson, 1980). The current version of ARSPIQ was developed on an Amdahl 470V/6-II and an Amdahl 470V/8 operated by the Data Processing Center of Texas A&M University. The version of FORTRAN used should be compatible with most FORTRAN IV or FORTRAN 77 compilers.

A User's Guide to ARSPIQ: The Univariate Time Series Aanlysis Program for Autoregressive Spectral Information Quantile Identification

1. Introduction

The ARSPIQ (AutoRegressive SPectral Information Quantile identification) program is a modified version of ARSPID, a univariate time series program in the TIMESBOARD Computing Library (Newton, 1983). ARSPIQ provides various diagnostics in the time and frequency domains to help one determine whether a time series is long, short, or no memory.

ARSPIQ is written in FORTRAN and consists of a main program and 58 subprograms, many of which are contained in the TIMESBOARD Computing Library. Many of the subprograms used in the quantile analysis are modified versions of those used by the ONESAM program (Parzen and Anderson, 1980). The current version of ARSPIQ was developed on an Amdahl 470V/6-II and an Amdahl 470V/8 operated by the Data Processing Center of Texas A&M University. The version of FORTRAN used should be compatible with most FORTRAN IV or FORTRAN 77 compilers.

The basic goal of ARSPIQ is to provide diagnostics to aid in modeling a univariate time series. ARSPIQ is not intended to be a modeling or forecasting program, however. The models produced by ARSPIQ are intended only as guides to formulate more complete, rigorous, or valid models. Our approach is to run ARSPIQ to obtain useful model building diagnostics and then to

employ ARSPID in the more formal model building stage, i.e., estimating parameters and checking residuals. With this goal in mind, ARSPIQ has been made as fully automatic as possible with "logical" defaults provided when input options are specified to be zero.

For a more complete discussion of the statistical functions mentioned in the next section, see Parzen (1981) and Parzen (1983). Also, for a complete discussion of the TIMESBOARD computing package and useful time series results, see Newton (1983).

2. Stages in an Analysis

2.1 Quantile Analysis

The original time series $Y(t)$, the periodogram $f_T(\omega)$, the correlogram $\rho_T(v)$, and the partial autocorrelations $\hat{a}_m(m)$ may be treated as data batches and exposed to a quantile analysis to gain further insight about these quantities. For a data batch $\{X(t),\ t=1,\dots,T\}$, one defines the informative quantile function $IQ(u)$ by

$$IQ(u) = \frac{X(t) - X_{50}}{2(X_{75} - X_{25})} \quad , \quad u = \frac{2t-1}{2T}, \qquad t = 1,\dots,T \quad ;$$

where X_{50} is the median and X_{75} and X_{25} are the upper and lower quartiles respectively. Plots of $IQ(u)$ are useful in describing a data batch and are primarily used in this work to diagnose probability characteristics. In ARSPIQ, the informative quantile function

for the original time series is used in the remaining analyses since it serves to "normalize and detrend" the series analogously to subtracting the mean and dividing by the standard deviation.

For a specified null value for the density-quantile function $fQ(u) \equiv f(Q(u))$ where f is a probability density and Q is the corresponding quantile function, and for a raw estimate of the quantile density q defined to be the derivative of Q, one computes the weighted spacings

$$\tilde{d}(u) = f_o Q_o(u)\ \tilde{q}(u)/\tilde{\sigma}_o$$

where

$$\tilde{\sigma}_o = \int_o^1 f_o Q_o(u)\ \tilde{q}(u)\ du \quad .$$

The weighted spacings $\tilde{d}(u)$ hence comprise a probability density and $\tilde{\sigma}_o$ is treated as an estimate of scale (compare to standard deviation). ARPSIQ takes

$$f_o Q_o(u) = \phi(\Phi^{-1}(u))$$

for the normal case where ϕ is the standard normal pdf and Φ^{-1} is the standard normal quantile function; and

$$f_o Q_o(u) = 1-u$$

for the exponential case. When a data batch comes from the specified null distribution, $\tilde{d}(u)$ oscillates about a uniform value of 1 and the cumulative weighted spacings

$$\tilde{D}(u) = \int_0^u \tilde{d}(v)\, dv \quad , \quad 0 \le u \le 1$$

oscillates about the identity line. ARSPIQ provides a plot of $\tilde{D}(u)$ as a goodness-of-fit check. For independent normal data, the sample autocorrelations and partial autocorrelations will also be normally distributed while the raw periodogram will have an exponential distribution, hence the normal and exponential cases are automatically checked where appropriate by ARSPIQ.

A quantile analysis also produces some useful descriptive statistics, but discussion of these will be withheld until the appropriate time.

2.2 Frequency Domain Analysis

The raw periodogram $f_T(\omega)$ is computed as a Fourier Transform of the time series and then normalized to integrate to one. Specifically,

$$f_T\left(\frac{k}{Q}\right) = (A_k^2 + B_k^2)\hat{\sigma}^2/T \quad , \quad k=1,\ldots,Q$$

where $\hat{\sigma}^2$ is the sample variance of the informative quantile of the data and A_k and B_k are the Fourier coefficients evaluated

at frequency k/Q. (A_k is the amplitude of the cosine term and B_k is the amplitude of the sine term at frequency k/Q when a sinusoid is fit to the data set. See Newton (1983).) As mentioned previously, white noise data produces a periodogram that has an exponential distribution, and hence a quantile analysis of $f_T(\omega)$ uses $f_o Q_o(u) = 1-u$.

As discussed in Parzen (1983), the key descriptive statistics for the raw periodogram are the median, variance, and spectral log range (or dynamic range) defined to be

$$\mathrm{SPLR} = \max_{\omega} \log f_T(\omega) - \min_{\omega} \log f_T(\omega) \quad .$$

Large values of periodogram variance or SPLR indicate moderate to long memory, while small values of periodogram median have the same interpretation. For normal white noise, the median is $\log 2 \doteq .693$, the variance is one, and SPLR = 0.

A nonparametric periodogram is also computed and is labeled "Local Quantile Periodogram" on the output. For a neighborhood length NQMP, a batch of periodogram ordinates of length NQMP produces a lower quartile, median, and upper quartile value. These values are labeled G25, G50, and G75 and are associated with the median frequency in the neighborhood. G50 is interpreted as a nonparametric smoothed estimate of the spectral density while G25 and G75 serve as "confidence limits." The log of these values is plotted.

A Parzen window estimator using truncation point M = {largest even integer $\leq$ (T/2)} also provides a smoothed periodogram. This estimate of a spectral density is used in creating the cepstral correlations described later. Parametric spectral estimators are also computed and will be described later. These estimators will have SPLR computed as well as a quantity known as the delta memory function defined by

$$\delta_k = \frac{1}{k} \sum_{j=1}^{k} \log f\left(\frac{j+m}{Q}\right) - \log f\left(\frac{k+1+m}{Q}\right)$$

where m = 0 and m = Q/XSEAS with Q = NFREQS and XSEAS specified by the user. By default, NFREQS is chosen large enough for the FFT routine and XSEAS = 12. These two delta sequences are plotted on a scale from -3 to 3 and have apparent memory classification properties as discussed in Parzen (1983).

2.3 Time Domain Analysis

Besides the interpretation provided by Box and Jenkins (1970), the sample autocorrelation function (acf.) and sample partial autocorrelation function (p.acf.) provide added insight when viewed in the quantile domain.

The autocorrelation function (correlogram) $\rho_T(v)$ is computed as the Fourier transform of the raw periodogram. Besides the plot of $\rho_T(v)$ and its quantile function, one also examines the value of

$$\frac{1}{T}\sum_{v=1}^{T} \rho_T^2(v) \quad (\text{or } \frac{1}{M}\sum_{v=1}^{M} \rho_T^2(v) \quad , M \leq T/2).$$

Small values of this quantity indicate no memory or short memory, while values larger than say 0.1 typically indicate long memory.

The partial autocorrelation function is given by the value of $\hat{a}_m(m)$ estimated via the Yule-Walker equations for the general AR(m) model denoted by

$$Y(t) + \alpha_m(1)\ Y(t-1) + \ldots + \alpha_m(m)\ Y(t-m) = \varepsilon(t)$$

where $\varepsilon(t)$ is white noise and $\hat{a}_m(m)$ estimates $\alpha_m(m)$.

A second partial autocorrelation function is computed using the Burg algorithm which appears to provide better estimates than the Yule-Walker equations when roots of the characteristic polynomials are near the unit circle (near nonstationarity).

Only the Yule-Walker estimates are exposed to a quantile analysis with normal $f_o Q_o$. The Burg estimate3 are primarily intended to produce an alternate approximating AR model for ARMA select modeling purposes and spectral analysis. Of interst is the number of values of the IQ function for the p.acf. that are greater than one in absolute value. Outliers in both the acf. and p.acf. data batches indicate moderate to long memory.

2.4 Diagnostic Modeling

Five parametric models are provided to give insight into the nature of the time series examined. Spectral densities are computed for four of the models and residual variances are stated for all five. In all cases the parametric models are actually treated as approximating models and hence are really nonparametric in nature. This suggests that ARSPIQ should not be used to provide THE model for a time series although one of the five models examined may be considered appropriate by some criterion.

The first two models considered are approximating autoregressive models with orders given by the first and second relative minima of the CAT function. Values of AIC and log residual variance are also provided in this AR modeling stage. For the best order model, a spectral analysis is carried out with computation of SPLR and δ_k and display of the various spectral quantities. Only the coefficients and residual variance (RVAR) are produced for the second best order model.

A second modeling stage actually produces five models, but only three are examined in detail. This second stage is primarily intended to suggest subset ARMA models to help better understand the nature of the series. In many cases the ARMA models are merely AR models mimicking the previous CAT derived ones, and hence little new information may be provided at this stage.

For subset ARMA modeling, the default strategy is based upon the Burg sample p.acf., but a second strategy based upon cepstral correlations may be requested by the user. The p.acf. obtained via the Burg algorithm is fed to a recursion algorithm which produces AR(p) coefficients where p is chosen to be the larger of the best and second best order as determined by CAT. Inverse autocorrelations are also computed and displayed. A spectral analysis is then performed on this AR model. The model is then inverted to an infinite MA and truncated to order M (NCOVM) corresponding to the number of computed autocorrelations. The residual variance and coefficients of this model are input to a select regression routine to produce an appropriate covariance matrix. Along with cross-correlation values, a value called PVH is also displayed. This is analogous to the prediction variance horizon defined in Parzen (1981) but is actually 1-PVH(h). For the infinite MA model

$$Y(t) = \varepsilon(t) + \beta_1 (t-1) + \beta_2\varepsilon(t-2) + \ldots$$

we define

$$PVH(h) = \sigma_\infty^2 (1 + \beta_1 + \ldots + \beta_h)$$

where σ_∞^2 is the normalized mean square prediction error (innovation variance) estimated by RVAR for the Burg AR model.

The horizon HOR is the smallest value of h for which $PVH(h) \geq .95$. As a rule of thumb, HOR = 0 implies no memory, HOR/(AR order) $\geq$ 4 implies long memory, and values in between have specific interpretations as suggested by Parzen (1981). The select regression procedure automatically picks a diagnostic ARMA model displaying various criterion values, RVAR, and spectral density.

A second select strategy is based upon obtaining truncated infinite MA coefficients directly from the cepstral correlations. The cepstral correlations are essentially the Fourier transform values of log spectral density. We define cepstral correlations by

$$\psi(v) = \int_0^1 \log f(\omega)\ e^{2\pi i\omega v}\ d\omega$$

where $f(\omega)$ is estimated by the Parzen window estimate of the spectral density mentioned earlier. A recursion algorithm provides the MA coefficients and may be expressed as

$$(n+1)\ \beta(n+1) = \sum_{k=0}^{n} (k+1)\ \psi(k+1)\ \beta(n-k).$$

Truncating to order M, we proceed as before to the ARMA select stage. In addition, an estimate of σ_∞^2 based upon the formula

$$\log \sigma_\infty^2 = \int_0^1 \log f(\omega)\ d\omega$$

is provided using the fact that the right hand side is actually $\psi(0)$ and hence

$$\sigma_\infty^2 = \exp\ [\psi(0)] \quad .$$

This value is labeled "SIGMA INFINITY SQUARED" on the output.

Note that the values of RVAR may also be treated as estimates of σ_∞^2. For the cepstral correlation truncated MA model, the residual variance is

$$\mathrm{RVAR} = 1/(1 + \beta_1^2 + \ldots + \beta_M^2).$$

The AR residual variances are obtained from the Yule-Walker equations, and the ARMA residual variances are obtained from the suitable element in the covariance matrix after a series of SWEEP operations has been performed.

3. Input Options

One of the primary goals of ARSPIQ is to make a complete memory analysis of a time series as automatic as possible. Hence, when a zero is specified several options will default to "acceptable" values. The input to ARSPIQ is described below. All values should be right justified.

3.1 First card of a run (in I5 Format)

Cols	Quantity	Purpose
1-5	NANS	Number of data sets to be analyzed

3.2 Second card of a run [in (6I5, 2F5.Ø) Format]

Cols	Quantity	Purpose
1-5	NTAPE	File number where data set to be analyzed resides (e.g., if NTAPE = 11, specify FT11FØØ1 DD card describing data file) Default: NTAPE = 5, data follows card 2 of input options.
6-10	IRWND	1 if NTAPE to be rewound before being read, 0 otherwise. Default: IRWND = 0.
11-15	NCOVM	Number of autocorrelations to compute. NCOVM $\leq$ 250. Default: NCOVM = (sample size)/2
16-20	NFREQS	Number of equally spaced frequencies at which to compute spectral density. 1152 $\geq$ NFREQS $\geq$ (sample size) + NCOVM. Default: Determined by subroutine FNFREQ
21-25	IOPTMX	2 if select ARMA modeling desired, 3 is ARMA modeling using cepstral correlations desired, 4 for both, 1 otherwise. Default: IOPTMX = 2.

Cols	Quantity	Purpose
26-30	NQMP	Length of neighborhood in computation of local quantile periodogram. If no local quantile analysis is desired, specify NQMP = -1. Default: NQMP = (sample size)/XSEAS
31-35	XSEAS	Seasonal period, e.g., 12 for monthly data. Default: XSEAS = 12.
36-40	REXP	Power or log transformation. Y(t) replaced by Y(t)**REXP if 0 REXP>1. If REXP = Ø. or 1. on input, no transformation is performed. If REXP 1., a log transformation is produced.

3.3 ARSPIQ data sets

ARSPIQ employs ARSPID data set conventions. All time series data sets must be stored in standard format as described below.

Card 1: A data set title (anywhere in the 80 columns)

Card 2: Columns 1-5: Sample size right justified

Columns 10-29: A FORTRAN format statement describing how the data has been entered on the remaining cards, e.g., (8F1Ø.Ø).

Card 3,4,...: The data entered in the format specified on Card 2.

3.4 JCL for executing ARSPIQ

```
// Job Card
//* JES3 Control Cards
//PROCLIB DD DSN=USR.R579.TW.PROCLIB,DISP-SHR
// EXEC ARSPIQ
//SYSIN DD *
   --input option cards --
   --//FTnnFØØ1 cards describing input files.
```

4. Sample Output

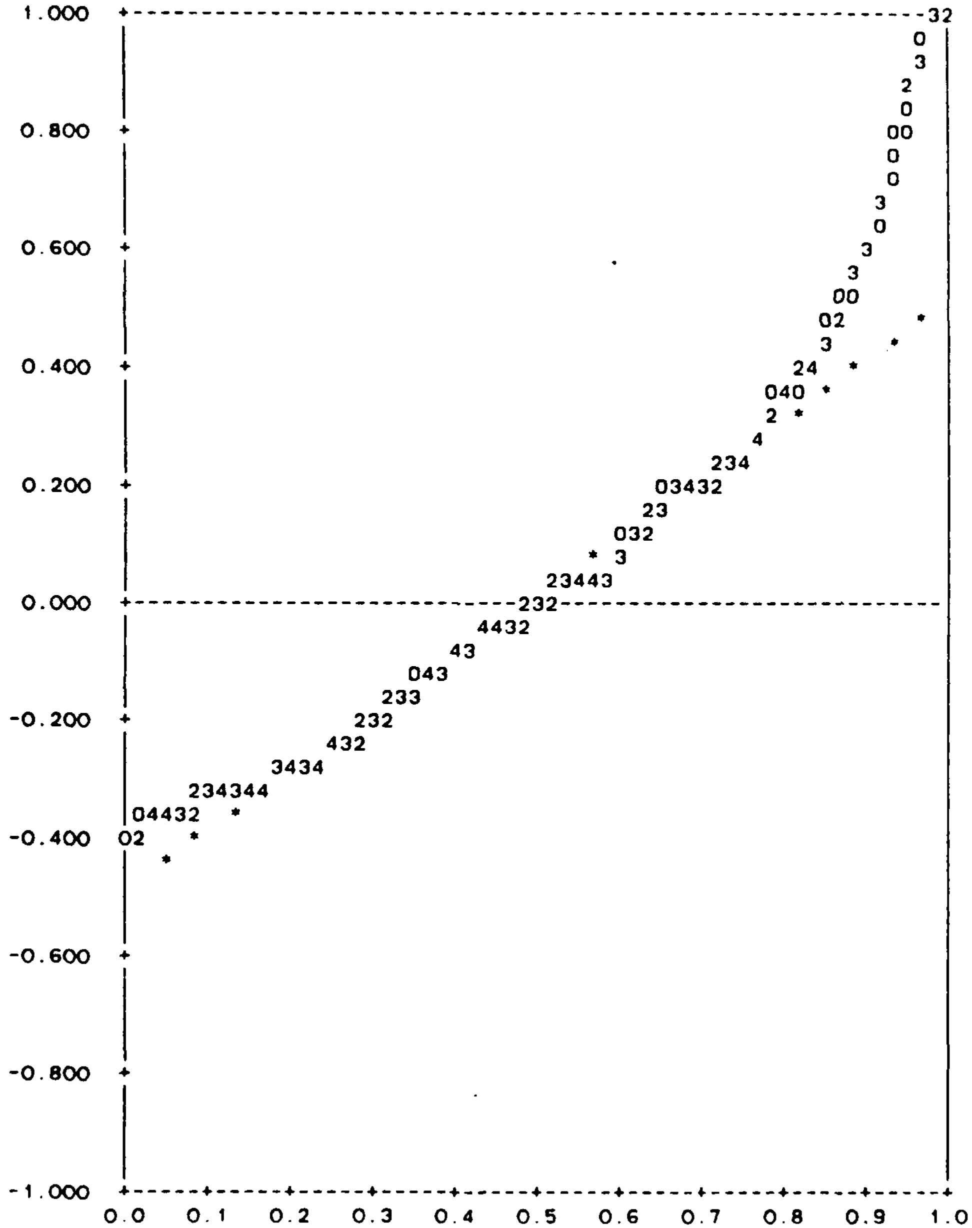
Following is an abbreviated output from ARSPIQ analyzing the Wolfer Sunspot numbers. The JCL that generated this run is given by:

```
//ARSPIQ JOB (R579,006C,S60,10,TW),'WOODFIELD',MSGLEVEL=(1,0)
//*FORMAT PR,DDNAME=,DEST=XEROX,JDE=JFMT1,FORMS=D100
//*LEVEL 1
//PROCLIB DD DSN=USR.R579.TW.PROCLIB,DISP=SHR
// EXEC ARSPIQ
//SYSIN DD *
    1
   13     0     0     0     4     0   11.   1.0
//FT13F001 DD DSN=USR.R579.TW.TSDATA(WOLFER),DISP=SHR,LABEL=(,,,IN)
```

```
*****************************************************************************
*                                                                           *
*   ARSPIQ - AUTOREGRESSIVE SPECTRAL INFORMATION QUANTILE IDENTIFICATION    *
*                                                                           *
*   EMANUEL PARZEN, TERRY J. WOODFIELD, AND H. JOSEPH NEWTON                *
*   DEPARTMENT OF STATISTICS, TEXAS A&M UNIVERSITY,                         *
*   COLLEGE STATION, TEXAS 77843                                            *
*   JULY 1983                                                               *
*                                                                           *
*****************************************************************************


C   WOLFERS SUNSPOT NUMBERS, 1749-1963                              WOLFER
          INPUT TAPE =  13    IRWND =    0        LENGTH OF SERIES =   215
          NCOVM = 106         NFREQS =  384       XSEAS =  11.00
          NQMP =  19          REXP =  1.000
```

C WOLFERS SUNSPOT NUMBERS, 1749-1963
INFORMATIVE QUANTILE - ORIGINAL DATA

WOLFER



ABSCISSA IS U , ORDINATE IS IQ(U)

U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.95000	0.99000
IQ(U)	-0.38550	-0.35821	-0.33531	-0.24905	0.25095	0.60095	0.86584	1.32511

FULLY NON-PARAMETRIC ANALYSIS
****

C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER
ORIGINAL DATA

DESCRIPTIVE STATISTICS
****

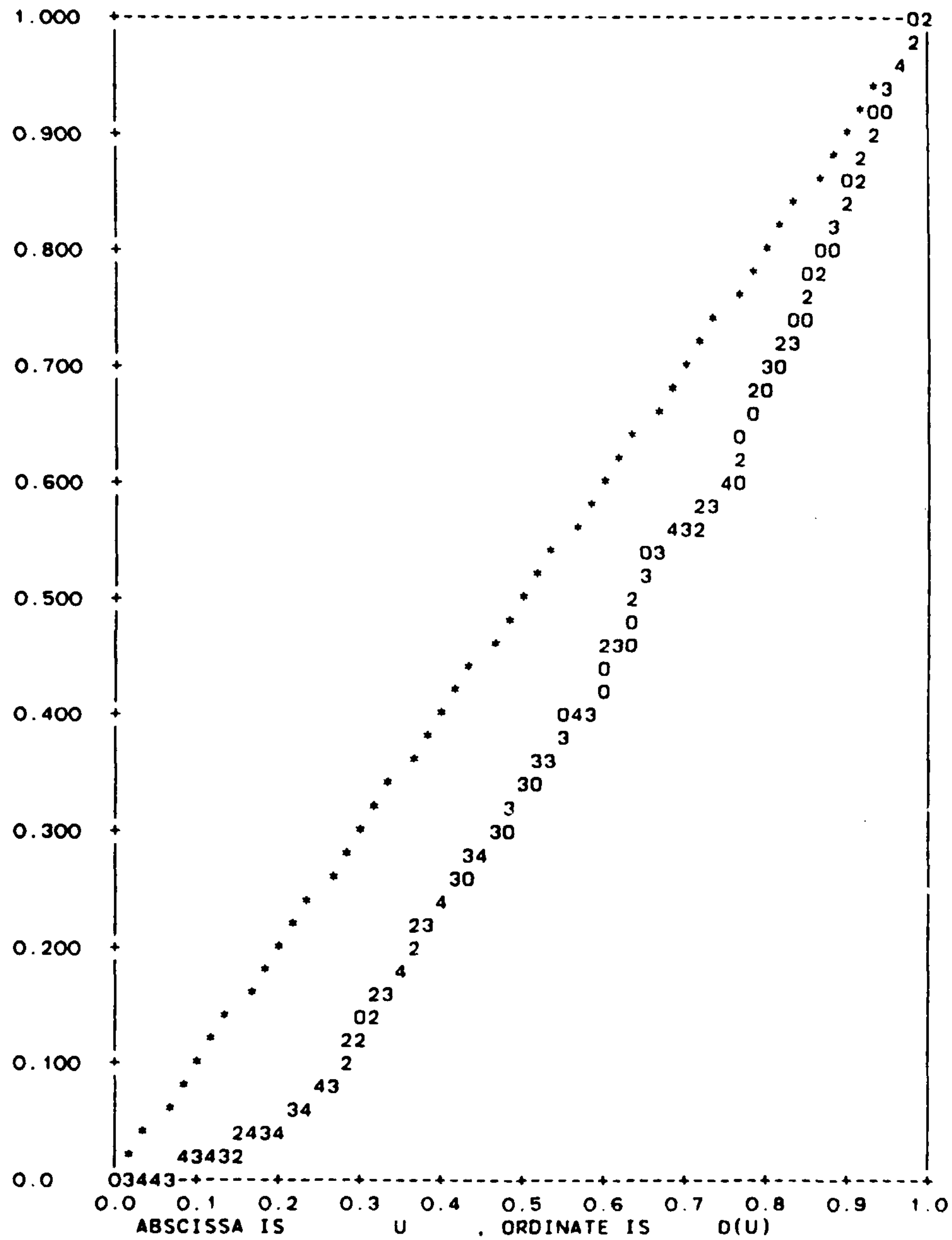
SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE
215	15.70	41.80	68.10	52.40

SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
3970.	49.20	1556.	39.44	.7065E-01	.3764	-.9772

AV. LOG SPACINGS	AV. LOG W. SPACINGS	AV. LOG HYP. FQ	SIGMA ZERO	LOG SIGMA ZERO
-.80315173E-01	-.45842719	-1.3995142	.36008394	-1.0214176

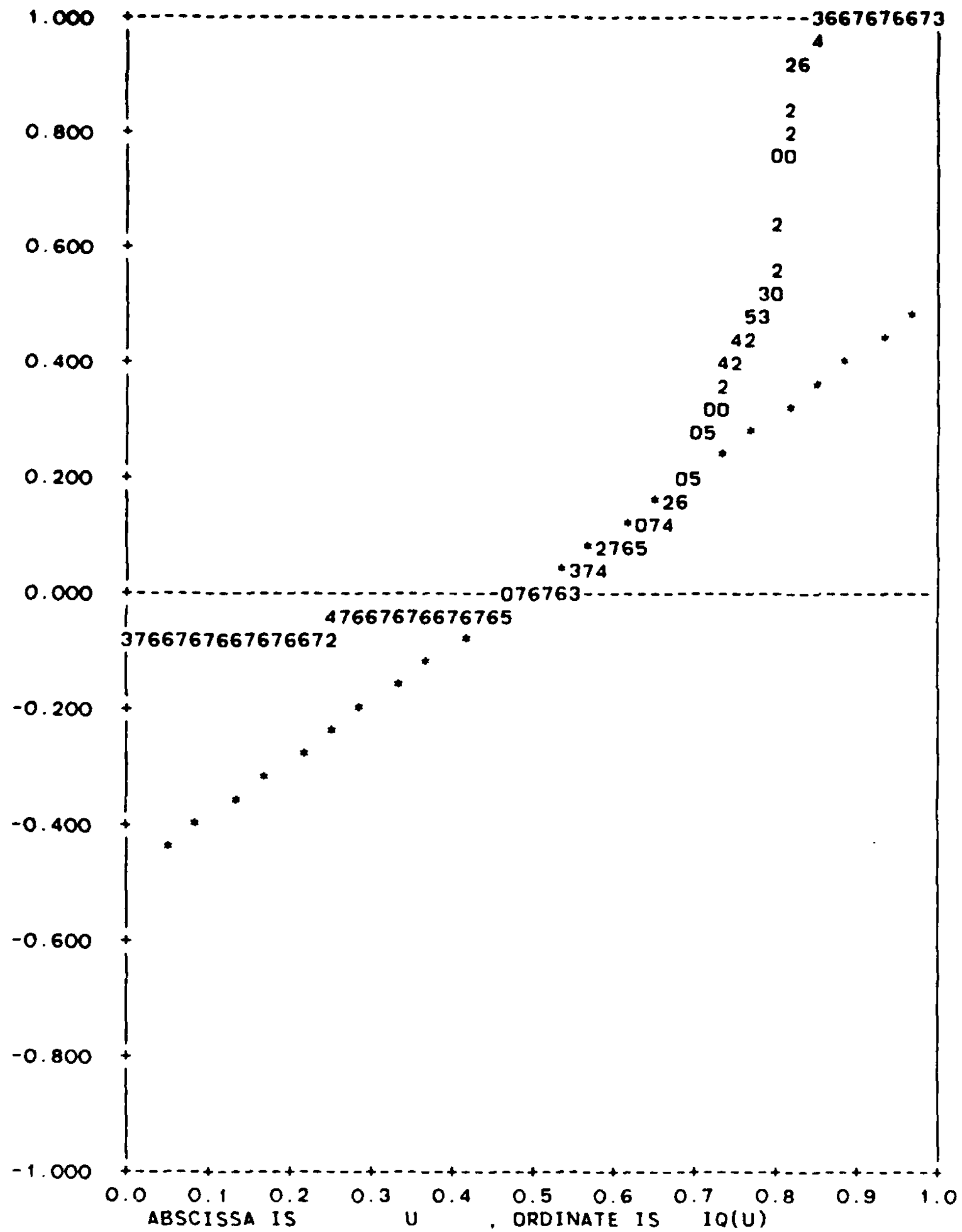
RAW DISTRIBUTION D(U) NORMAL CASE

D+ = 0.0870 AT U = 0.9930 . D- = -2.7392 AT U = 0.5884



C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER

INFORMATIVE QUANTILE - RAW PERIODOGRAM



U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.95000	0.99000
IQ(U)	-0.08192	-0.07895	-0.07583	-0.05973	0.44027	2.85387	6.60207	22.14940

FULLY NON-PARAMETRIC ANALYSIS

C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER
RAW PERIODOGRAM

DESCRIPTIVE STATISTICS

SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE
384	.2133E-01	.7664E-01	.4844	.4630

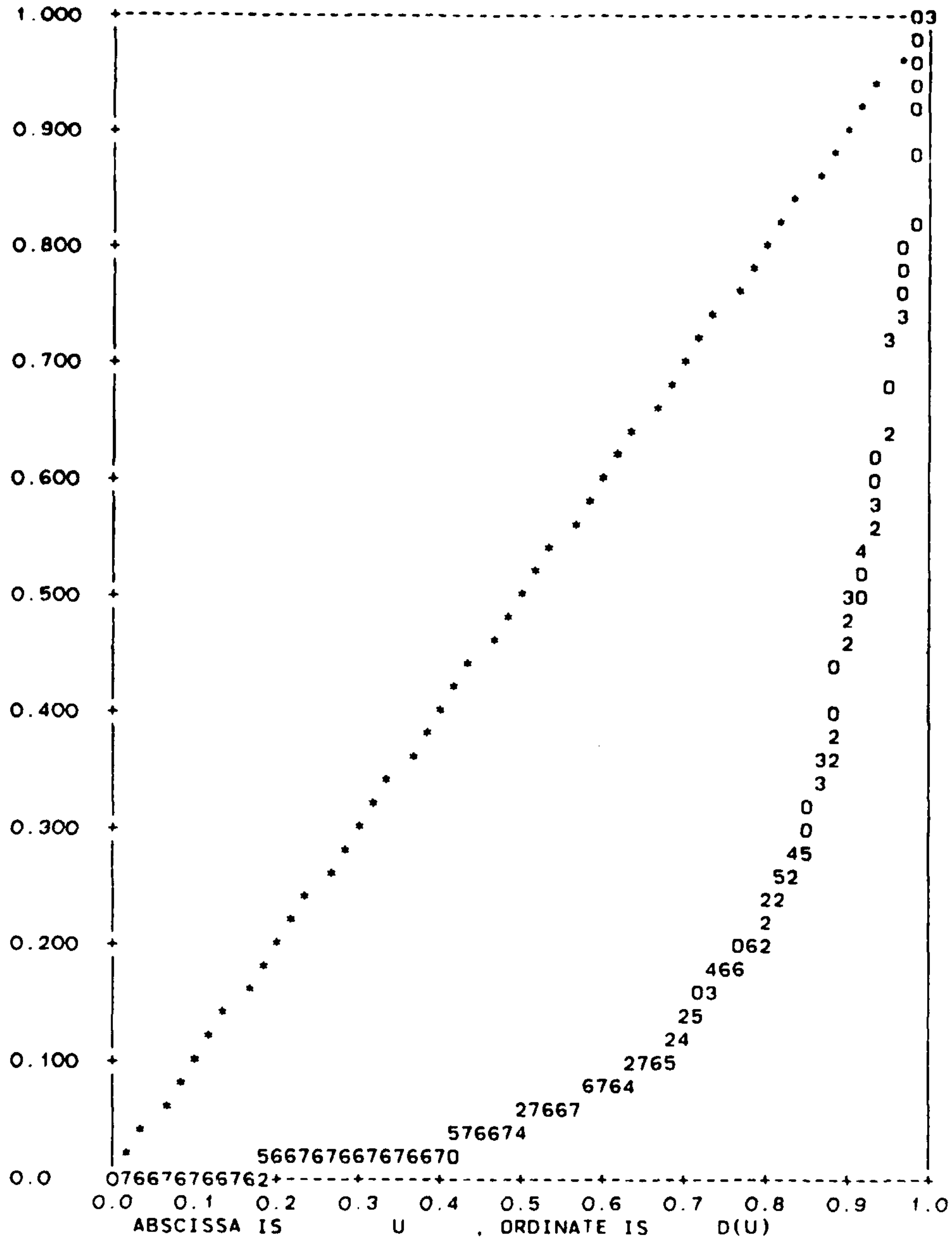
SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
9.867	1.000	8.890	2.982	.9971	3.220	1.169

MAXIMUM JUMP FOR QUANTILE FUNCTION = 921.1580 CORRESPONDING TO Q(U) = 9.96218

I	U	Q
375	0.975	9.96218
376	0.978	9.96220
377	0.980	14.40501
378	0.983	14.40503
379	0.986	16.70163
380	0.988	16.70164
381	0.991	20.58804
382	0.993	20.58804
383	0.996	21.89549
384	0.999	21.89554

AV. LOG SPACINGS	AV. LOG W. SPACINGS	AV. LOG HYP. FQ	SIGMA ZERO	LOG SIGMA ZERO
-.74432033	-1.8252487	-1.0015879	1.0825577	.79326451E-01

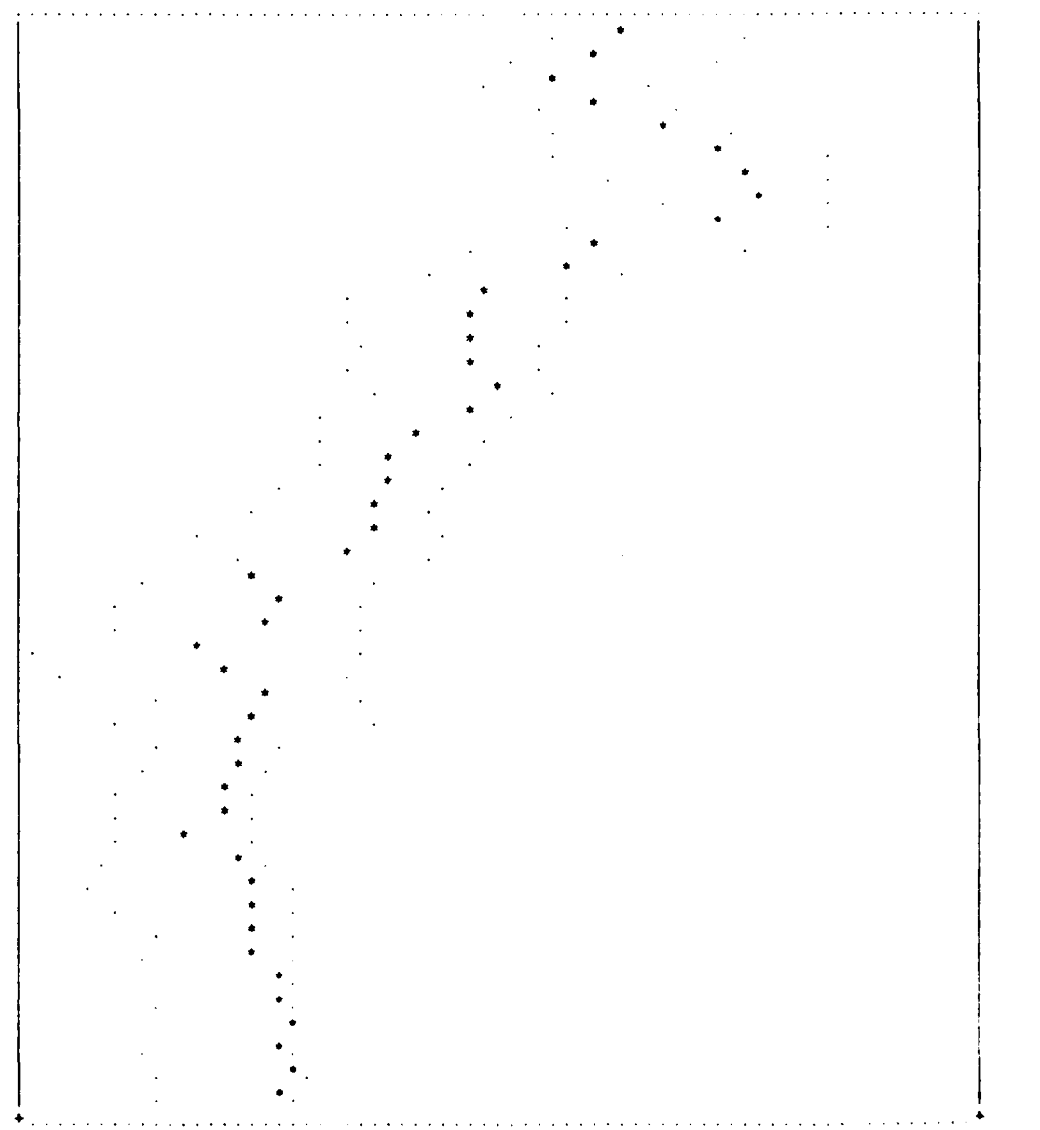
RAW DISTRIBUTION D(U) EXPONTAL CASE
D+ = 0.0776 AT U = 0.9935 . D- = ******* AT U = 0.7956



LOCAL QUANTILE PERIODOGRAM: K = 19

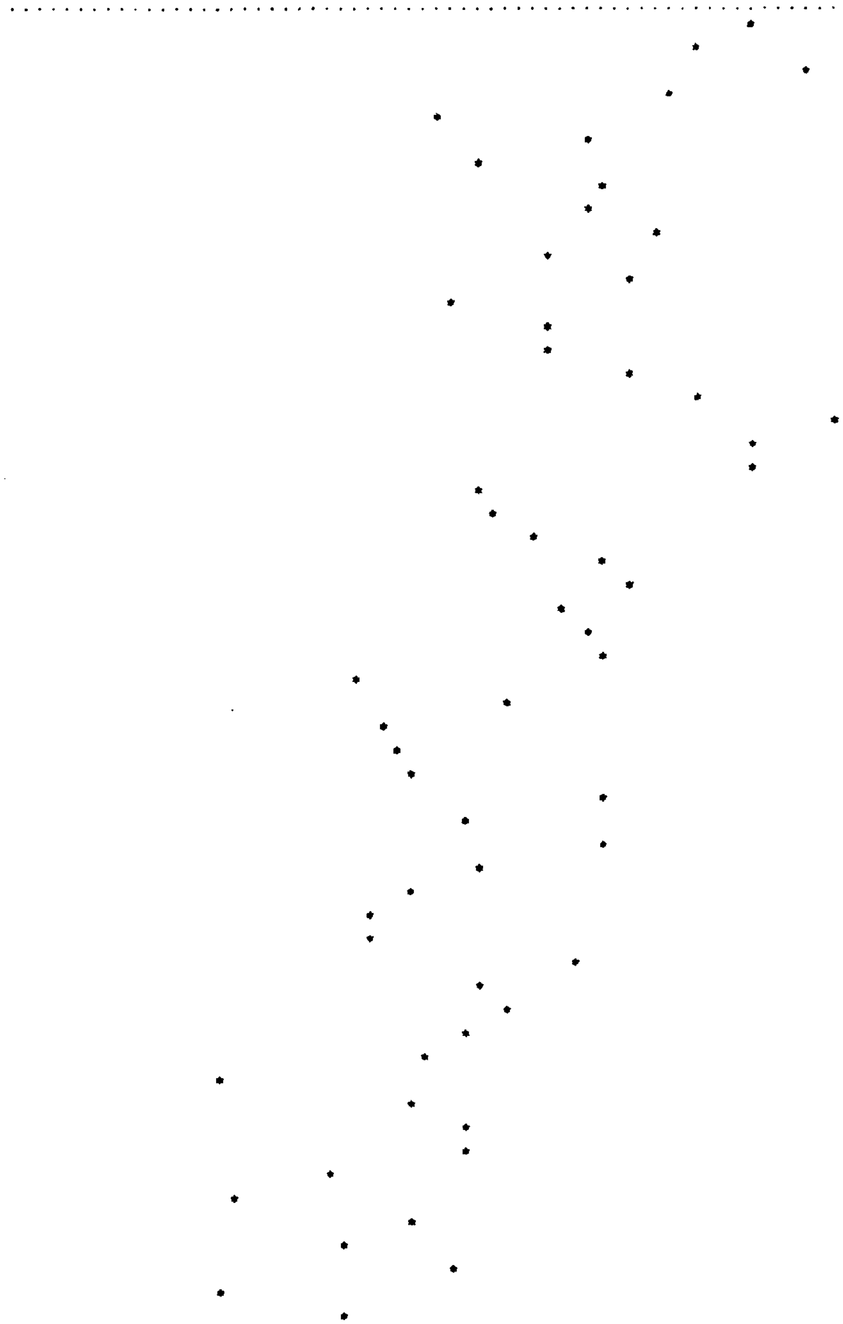
RAW PERIODOGRAM

U	G25	G50	G75
0.0026	0.4844	1.0109	3.4683
0.0137	0.3255	0.7970	2.5850
0.0248	0.2619	0.5201	1.3799
0.0359	0.4360	0.7326	1.6044
0.0470	0.4850	1.4335	2.9212
0.0582	0.5235	2.5507	7.1950
0.0693	0.8855	3.4208	8.0483
0.0804	1.5137	3.8399	8.0483
0.0915	0.5440	2.6421	7.9348
0.1026	0.2298	0.7373	3.4292
0.1137	0.1473	0.5361	0.9730
0.1248	0.0622	0.2654	0.5915
0.1359	0.0622	0.2204	0.5775
0.1470	0.0689	0.2204	0.4596
0.1582	0.0660	0.2204	0.4019
0.1693	0.0836	0.2716	0.5067
0.1804	0.0508	0.2052	0.3271
0.1915	0.0496	0.1207	0.2609
0.2026	0.0496	0.1003	0.2052
0.2137	0.0315	0.0924	0.1609
0.2248	0.0241	0.0869	0.1506
0.2359	0.0138	0.0812	0.1588
0.2470	0.0219	0.0605	0.1375
0.2582	0.0079	0.0242	0.0774
0.2693	0.0064	0.0319	0.0711
0.2804	0.0064	0.0291	0.0687
0.2915	0.0027	0.0140	0.0687
0.3026	0.0036	0.0172	0.0630
0.3137	0.0095	0.0281	0.0766
0.3248	0.0059	0.0233	0.0811
0.3359	0.0091	0.0205	0.0301
0.3470	0.0084	0.0198	0.0284
0.3582	0.0064	0.0177	0.0237
0.3693	0.0061	0.0175	0.0240
0.3804	0.0057	0.0123	0.0236
0.3915	0.0054	0.0216	0.0266
0.4026	0.0048	0.0236	0.0367
0.4137	0.0064	0.0230	0.0368
0.4248	0.0094	0.0237	0.0379
0.4359	0.0085	0.0247	0.0381
0.4470	0.0086	0.0321	0.0388
0.4582	0.0088	0.0324	0.0388
0.4693	0.0088	0.0345	0.0432
0.4804	0.0082	0.0324	0.0353
0.4915	0.0088	0.0345	0.0443
0.5026	0.0086	0.0324	0.0388

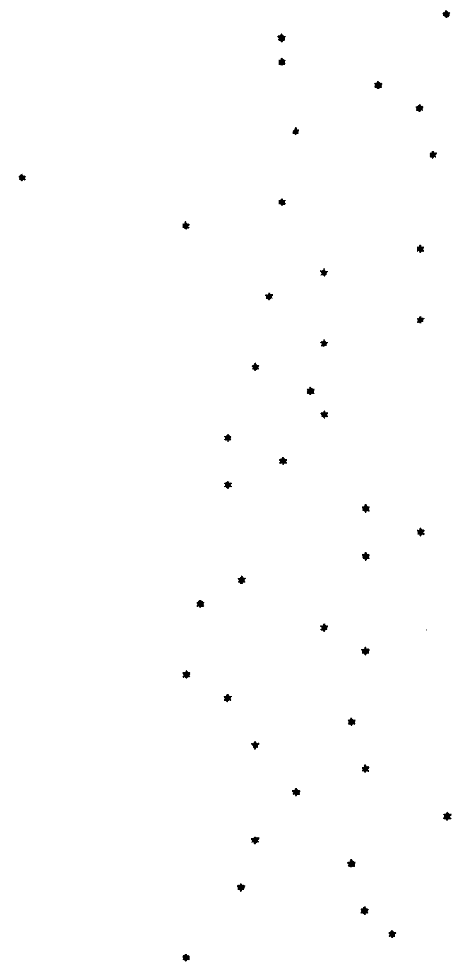


FOR PERIODOGRAM, SPLMIN = -6.00000 SPLMAX = 3.08628 R(O) = 0.14599
SPLR = 9.08628

FREQUENCY	PERIOD	SPEC
0.0	0.0	7.35124
0.00521	192.00005	3.46834
0.01042	96.00002	14.40504
0.01562	64.00002	2.71947
0.02083	48.00000	0.11259
0.02604	38.40001	0.80144
0.03125	32.00000	0.18917
0.03646	27.42857	0.92473
0.04167	24.00000	0.78488
0.04687	21.33333	2.21510
0.05208	19.20000	0.48457
0.05729	17.45454	1.32517
0.06250	16.00000	0.13735
0.06771	14.76923	0.52351
0.07292	13.71429	0.46361
0.07812	12.80001	1.51372
0.08333	12.00000	3.42081
0.08854	11.29412	21.89557
0.09375	10.66667	8.04830
0.09896	10.10526	7.19496
0.10417	9.60000	0.20100
0.10937	9.14286	0.22976
0.11458	8.72727	0.37931
0.11979	8.34783	0.94114
0.12500	8.00000	1.53267
0.13021	7.68000	0.53605
0.13542	7.38462	0.84667
0.14062	7.11111	0.97304
0.14583	6.85715	0.03836
0.15104	6.62069	0.26536
0.15625	6.40000	0.05312
0.16146	6.19355	0.06217
0.16667	6.00000	0.07664
0.17187	5.81819	0.92178
0.17708	5.64706	0.16320
0.18229	5.48572	0.97028
0.18750	5.33334	0.19370
0.19271	5.18919	0.08359
0.19792	5.05263	0.04502
0.20312	4.92308	0.04961
0.20833	4.80000	0.66983
0.21354	4.68293	0.20518
0.21875	4.57143	0.26090
0.22396	4.46512	0.15061
0.22917	4.36364	0.09238
0.23437	4.26667	0.00660
0.23958	4.17392	0.08692
0.24479	4.08511	0.15877
0.25000	4.00000	0.16091
0.25521	3.91837	0.02454
0.26042	3.84000	0.00814
0.26562	3.76471	0.07741
0.27083	3.69231	0.03474
0.27604	3.62264	0.13419
0.28125	3.55556	0.00636
0.28646	3.49091	0.03193

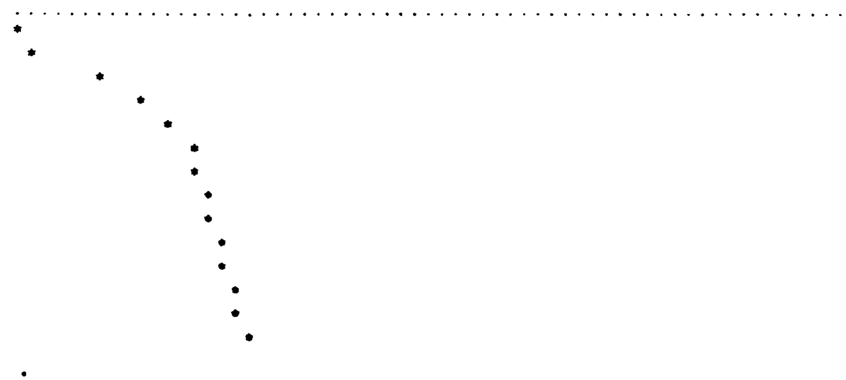


0.29167	3.42857	0.12415
0.29687	3.36842	0.01379
0.30208	3.31035	0.01279
0.30729	3.25424	0.04471
0.31250	3.20000	0.07264
0.31771	3.14754	0.01641
0.32292	3.09678	0.09815
0.32812	3.04762	0.00041
0.33333	3.00000	0.01256
0.33854	2.95385	0.00353
0.34375	2.90909	0.08348
0.34896	2.86567	0.02047
0.35417	2.82353	0.01048
0.35937	2.78261	0.08108
0.36458	2.74286	0.02307
0.36979	2.70423	0.00907
0.37500	2.66667	0.01734
0.38021	2.63014	0.02165
0.38542	2.59459	0.00642
0.39062	2.56000	0.01229
0.39583	2.52632	0.00599
0.40104	2.49351	0.03669
0.40625	2.46154	0.0774[illegible]
0.41146	2.43038	0.03894
0.41667	2.40000	0.00704
0.42187	2.37037	0.00429
0.42708	2.34146	0.02318
0.43229	2.31325	0.03761
0.43750	2.28571	0.00393
0.44271	2.25882	0.00679
0.44792	2.23256	0.03159
0.45312	2.20690	0.00845
0.45833	2.18182	0.03781
0.46354	2.15730	0.01529
0.46875	2.13333	0.11131
0.47396	2.10989	0.00864
0.47917	2.08696	0.03236
0.48437	2.06452	0.00725
0.48958	2.04255	0.03876
0.49479	2.02105	0.05391
0.50000	2.00000	0.00378

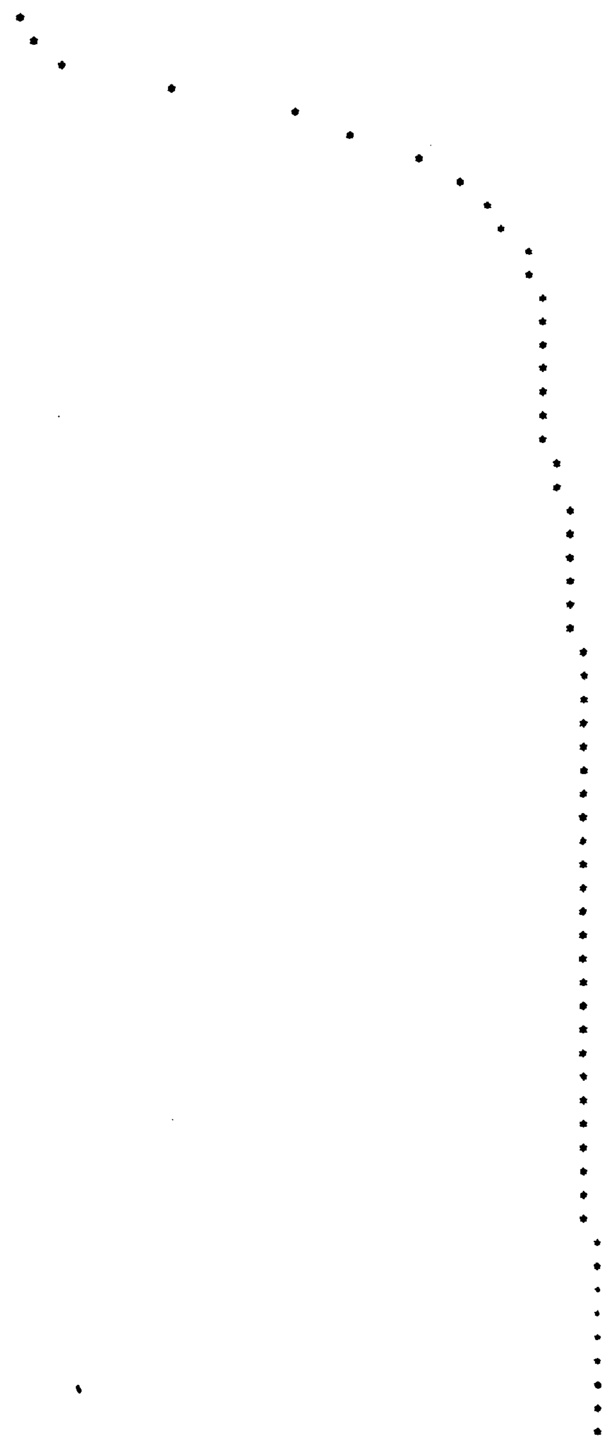


CUMULATIVE PERIODOGRAM :

FREQUENCY	PERIOD	SPEC
0.0	0.0	0.03757
0.00521	192.00005	0.06011
0.01042	96.00002	0.13889
0.01562	64.00002	0.18653
0.02083	48.00000	0.22121
0.02604	38.40001	0.24239
0.03125	32.00000	0.24583
0.03646	27.42857	0.25887
0.04167	24.00000	0.26370
0.04687	21.33333	0.27572
0.05208	19.20000	0.28071
0.05729	17.45454	0.28914
0.06250	16.00000	0.29248
0.06771	14.76923	0.30246



0.07292	13.71429	0.31957
0.07812	12.80001	0.34034
0.08333	12.00000	0.36650
0.08854	11.29412	0.50208
0.09375	10.66667	0.64843
0.09896	10.10526	0.70503
0.10417	9.60000	0.79141
0.10937	9.14286	0.84350
0.11458	8.72727	0.86506
0.11979	8.34783	0.89520
0.12500	8.00000	0.91740
0.13021	7.68000	0.92293
0.13542	7.38462	0.93019
0.14062	7.11111	0.93622
0.14583	6.85715	0.93671
0.15104	6.62069	0.94042
0.15625	6.40000	0.94129
0.16146	6.19355	0.94171
0.16667	6.00000	0.94323
0.17187	5.81819	0.95029
0.17708	5.64706	0.95416
0.18229	5.48572	0.96117
0.18750	5.33334	0.96355
0.19271	5.18919	0.96565
0.19792	5.05263	0.96705
0.20312	4.92308	0.96896
0.20833	4.80000	0.97497
0.21354	4.68293	0.97609
0.21875	4.57143	0.97769
0.22396	4.46512	0.97858
0.22917	4.36364	0.97931
0.23437	4.26667	0.98008
0.23958	4.17392	0.98103
0.24479	4.08511	0.98284
0.25000	4.00000	0.98382
0.25521	3.91837	0.98491
0.26042	3.84000	0.98542
0.26562	3.76471	0.98586
0.27083	3.69231	0.98615
0.27604	3.62264	0.98770
0.28125	3.55556	0.98774
0.28646	3.49091	0.98821
0.29167	3.42857	0.98920
0.29687	3.36842	0.98928
0.30208	3.31035	0.98935
0.30729	3.25424	0.98959
0.31250	3.20000	0.99000
0.31771	3.14754	0.99037
0.32292	3.09678	0.99102
0.32812	3.04762	0.99143
0.33333	3.00000	0.99151
0.33854	2.95385	0.99155
0.34375	2.90909	0.99212
0.34896	2.86567	0.99254
0.35417	2.82353	0.99275
0.35937	2.78261	0.99380
0.36458	2.74286	0.99394
0.36979	2.70423	0.99411
0.37500	2.66667	0.99421
0.38021	2.63014	0.99441
0.38542	2.59459	0.99455

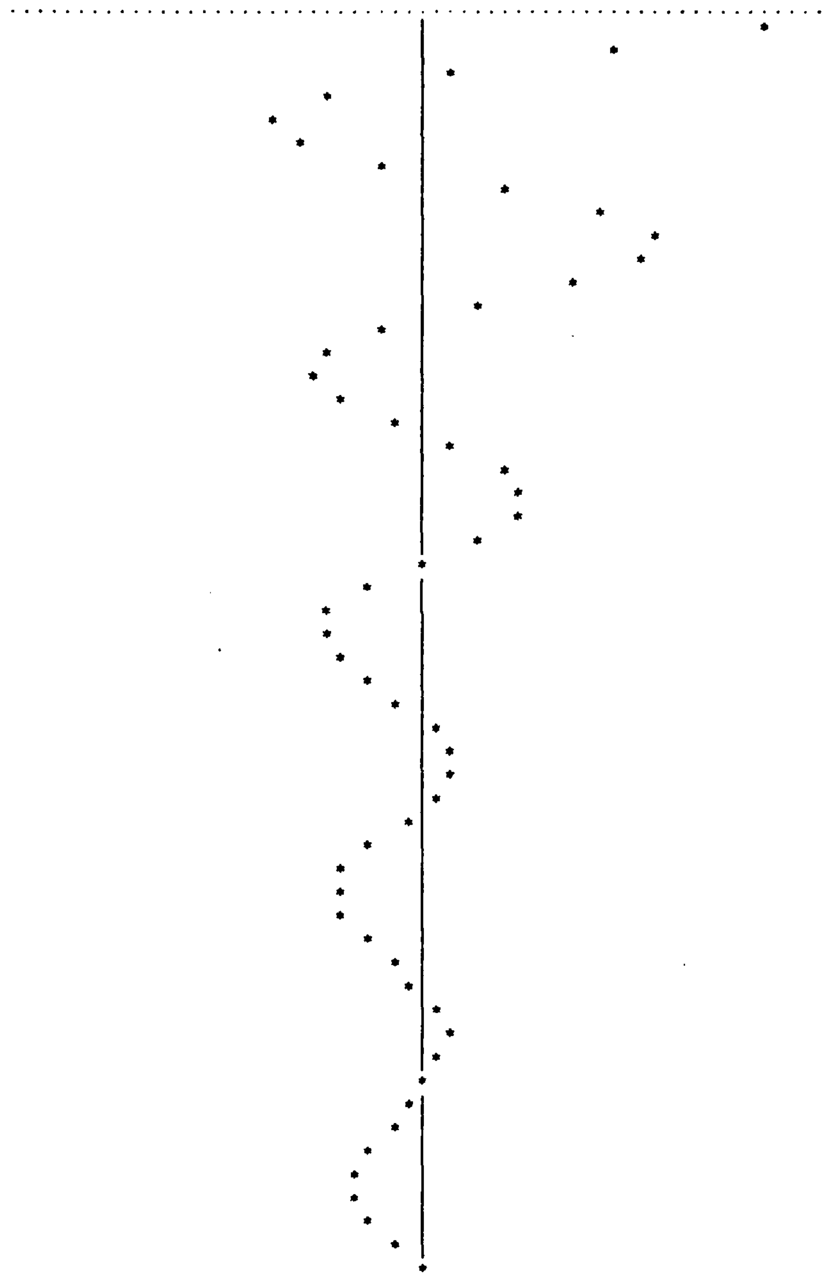


0.39062	2.56000	0.99465	*
0.39583	2.52632	0.99471	*
0.40104	2.49351	0.99501	*
0.40625	2.46154	0.99554	*
0.41146	2.43038	0.99575	*
0.41667	2.40000	0.99581	*
0.42187	2.37037	0.99584	*
0.42708	2.34146	0.99608	*
0.43229	2.31325	0.99639	*
0.43750	2.28571	0.99653	*
0.44271	2.25882	0.99669	*
0.44792	2.23256	0.99705	*
0.45312	2.20690	0.99713	*
0.45833	2.18182	0.99752	*
0.46354	2.15730	0.99768	*
0.46875	2.13333	0.99852	*
0.47396	2.10989	0.99879	*
0.47917	2.08696	0.99900	*
0.48437	2.06452	0.99908	*
0.48958	2.04255	0.99945	*
0.49479	2.02105	0.99991	*
0.50000	2.00000	1.00000	*

TIME DOMAIN ANALYSIS: CORRELOGRAM

R(O) = 0.1459852E+00

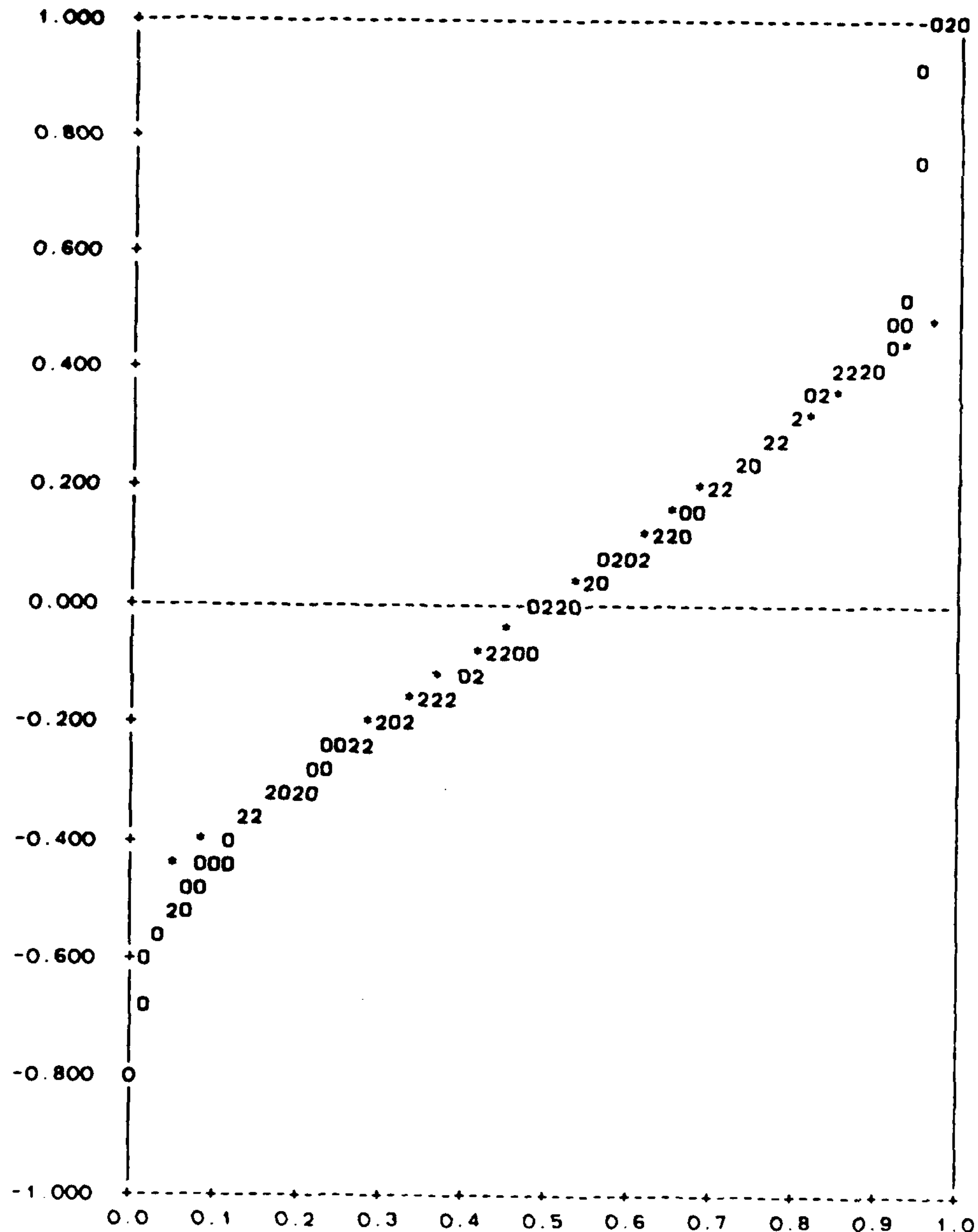
I	CORR
1	0.8221
2	0.4577
3	0.0637
4	-0.2258
5	-0.3531
6	-0.2974
7	-0.0957
8	0.1843
9	0.4354
10	0.5640
11	0.5328
12	0.3582
13	0.1179
14	-0.0972
15	-0.2268
16	-0.2544
17	-0.1873
18	-0.0673
19	0.0719
20	0.1851
21	0.2393
22	0.2246
23	0.1378
24	0.0076
25	-0.1250
26	-0.2223
27	-0.2473
28	-0.2022
29	-0.1331
30	-0.0558
31	0.0170
32	0.0730
33	0.0828
34	0.0451
35	-0.0377
36	-0.1273
37	-0.1886
38	-0.2121
39	-0.1918
40	-0.1428
41	-0.0820
42	-0.0212
43	0.0289
44	0.0544
45	0.0464
46	0.0122
47	-0.0244
48	-0.0748
49	-0.1223
50	-0.1600
51	-0.1623
52	-0.1298
53	-0.0702
54	0 0009



55	0.0728
56	0.1192
57	0.1307
58	0.1018
59	0.0220
60	-0.0816
61	-0.1557
62	-0.1774
63	-0.1416
64	-0.0530
65	0.0542
66	0.1456
67	0.1913
68	0.1761
69	0.1097
70	0.0159
71	-0.0776
72	-0.1373
73	-0.1485
74	-0.1080
75	-0.0193
76	0.0847
77	0.1675
78	0.2005
79	0.1842
80	0.1241
81	0.0495
82	-0.0299
83	-0.0810
84	-0.0918
85	-0.0432
86	0.0554
87	0.1584
88	0.2155
89	0.2259
90	0.1955
91	0.1292
92	0.0498
93	-0.0354
94	-0.0914
95	-0.0999
96	-0.0596
97	0.0132
98	0.1055
99	0.1718
100	0.1894
101	0.1603
102	0.0975
103	0.0244
104	-0.0354
105	-0.0648
106	-0.0626

C WOLFERS SUNSPOT NUMBERS, 1749-1963
INFORMATIVE QUANTILE - CORRELOGRAM

WOLFER



U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.95000	0.99000
IQ(U)	-0.78991	-0.51988	-0.43868	-0.24595	0.25405	0.42923	0.87669	1.74723

FULLY NON-PARAMETRIC ANALYSIS

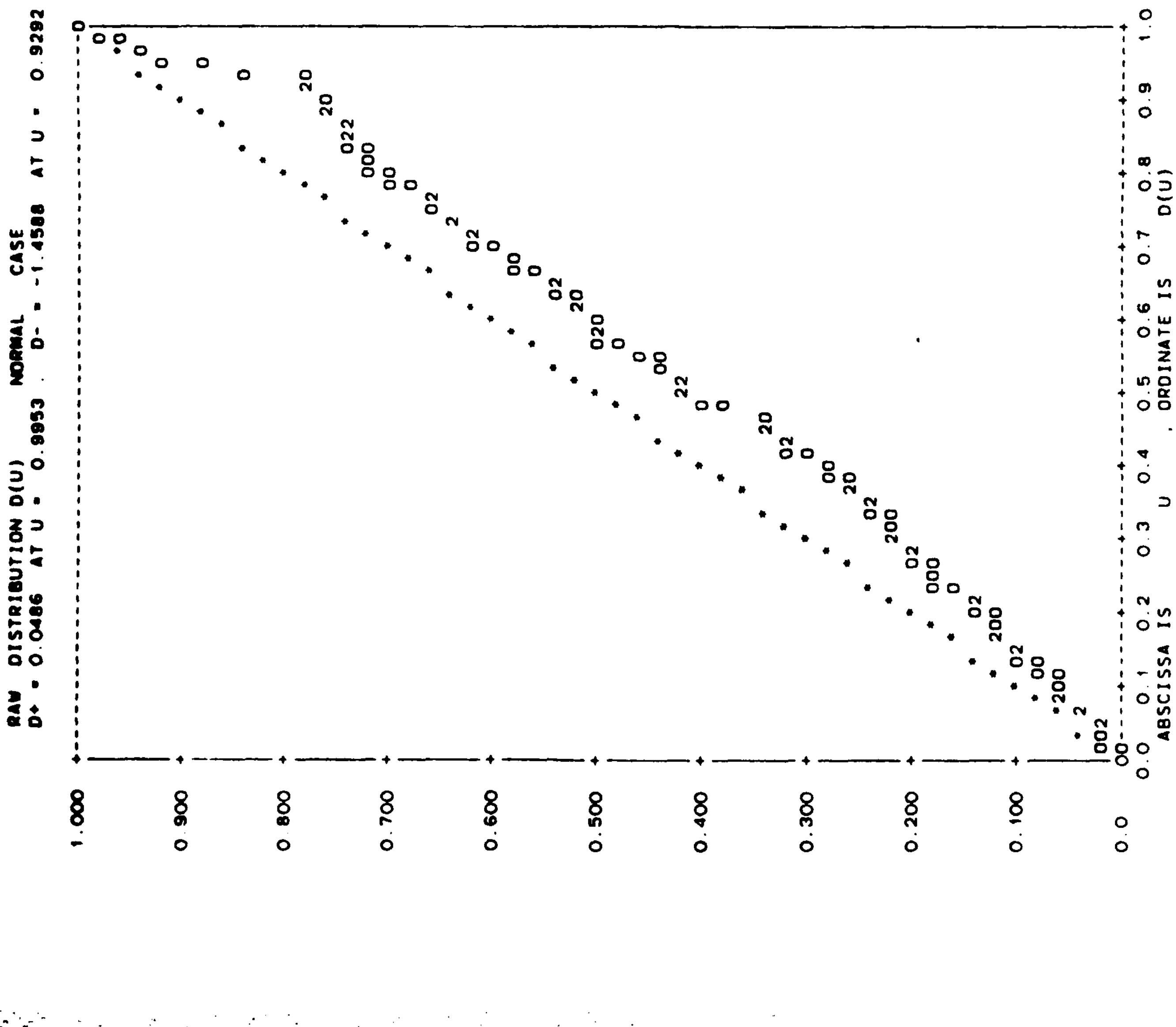
C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER
CORRELOGRAM

DESCRIPTIVE STATISTICS

SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE
106	-.1019	.9885E-02	.1254	.2273

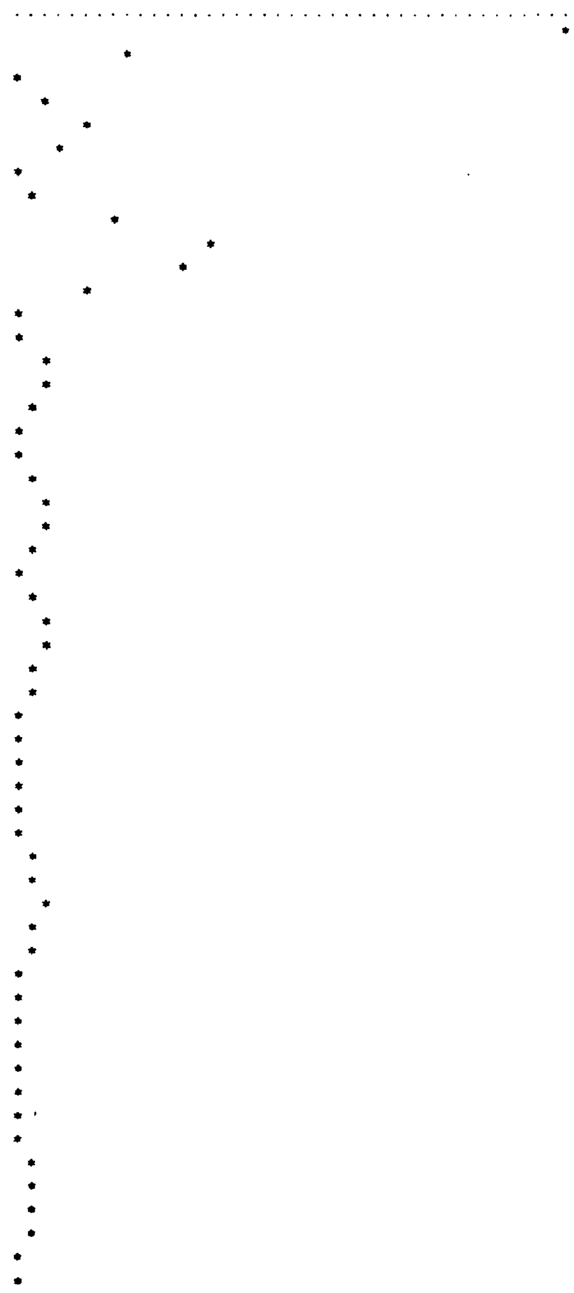
SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
.3487E-01	.2049E-01	.3478E-01	.1865	.2333E-01	.4103	-.8909

AV. LOG SPACINGS	AV. LOG W. SPACINGS	AV. LOG HYP. FQ	SIGMA ZERO	LOG SIGMA ZERO
.19785833	-.27064902	-1.3855381	.39970297	-.91703361


RAW DISTRIBUTION D(U) NORMAL CASE
D+ = 0.0486 AT U = 0.9953 . D- = -1.4588 AT U = 0.9292
1.000
0.900
0.800
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0.0
0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1.0
ABSCISSA IS U , ORDINATE IS D(U)

INFORMATION LAG FUNCTION: -0.5*LOG(1-RHO**2)

I	INFO
1	0.5634
2	0.1175
3	0.0020
4	0.0262
5	0.0666
6	0.0463
7	0.0046
8	0.0173
9	0.1051
10	0.1915
11	0.1670
12	0.0686
13	0.0070
14	0.0047
15	0.0264
16	0.0335
17	0.0179
18	0.0023
19	0.0026
20	0.0174
21	0.0295
22	0.0259
23	0.0096
24	0.0000
25	0.0079
26	0.0253
27	0.0315
28	0.0209
29	0.0089
30	0.0016
31	0.0001
32	0.0027
33	0.0034
34	0.0010
35	0.0007
36	0.0082
37	0.0181
38	0.0230
39	0.0187
40	0.0103
41	0.0034
42	0.0002
43	0.0004
44	0.0015
45	0.0011
46	0.0001
47	0.0003
48	0.0028
49	0.0075
50	0.0130
51	0.0133
52	0.0085
53	0.0025
54	0.0000

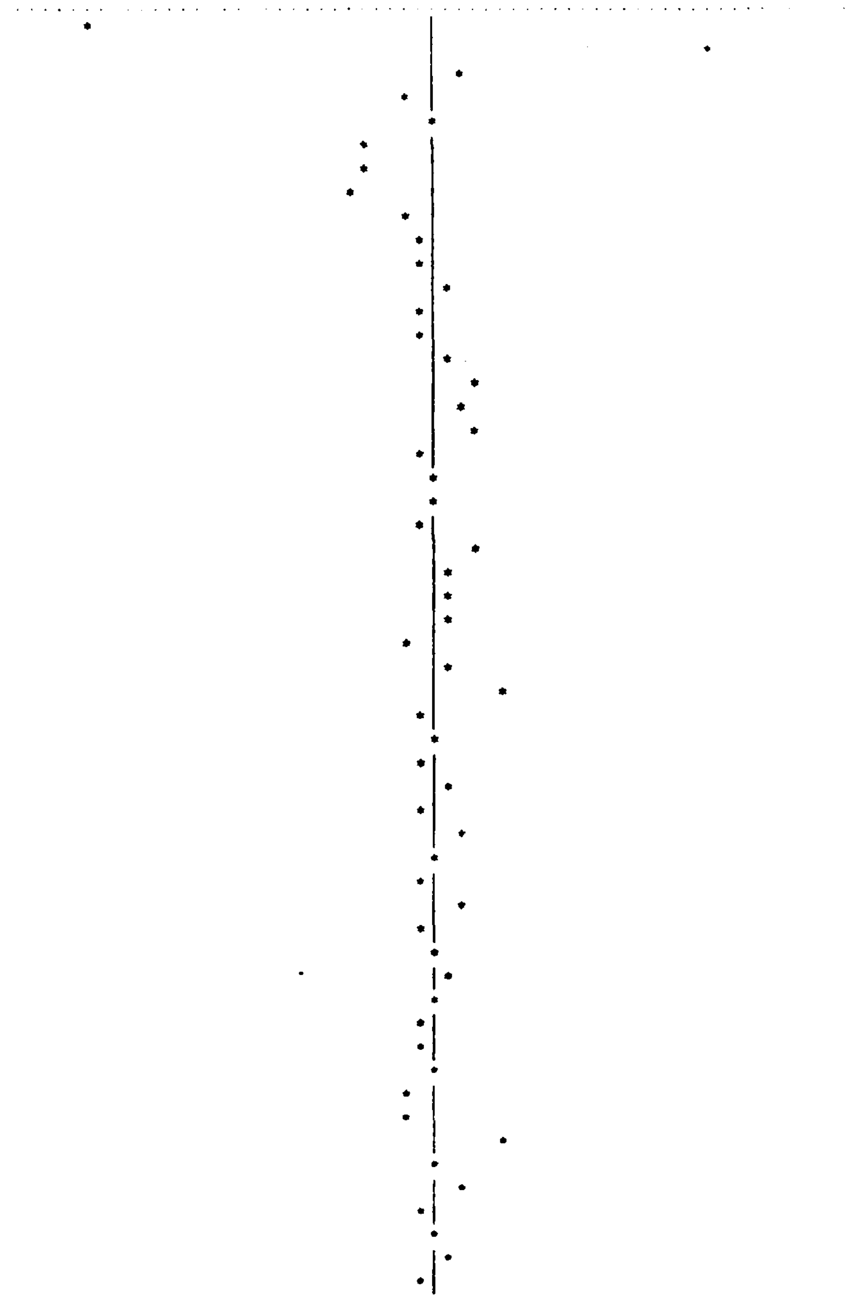


55	0.0027	*
56	0.0072	*
57	0.0086	*
58	0.0052	*
59	0.0002	*
60	0.0033	*
61	0.0123	*
62	0.0160	*
63	0.0101	*
64	0.0014	*
65	0.0015	*
66	0.0107	*
67	0.0186	*
68	0.0158	*
69	0.0060	*
70	0.0001	*
71	0.0030	*
72	0.0095	*
73	0.0112	*
74	0.0059	*
75	0.0002	*
76	0.0036	*
77	0.0142	*
78	0.0205	*
79	0.0173	*
80	0.0078	*
81	0.0012	*
82	0.0004	*
83	0.0033	*
84	0.0042	*
85	0.0009	*
86	0.0015	*
87	0.0127	*
88	0.0238	*
89	0.0262	*
90	0.0195	*
91	0.0084	*
92	0.0012	*
93	0.0006	*
94	0.0042	*
95	0.0050	*
96	0.0018	*
97	0.0001	*
98	0.0056	*
99	0.0150	*
100	0.0183	*
101	0.0130	*
102	0.0048	*
103	0.0003	*
104	0.0006	*
105	0.0021	*
106	0.0020	*

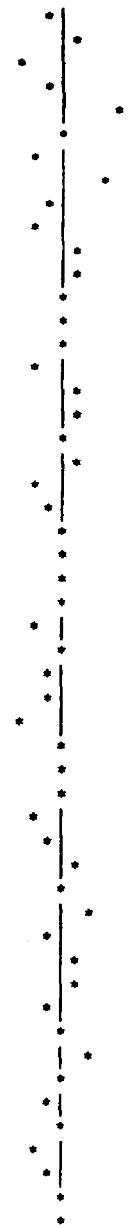
AR DESCRIPTION OF Y TILDA :

PARTIAL AUTOCORRELATIONS VIA THE YULE WALKER EQUATIONS:

I	PACF
1	-0.8221
2	0.6733
3	0.0698
4	-0.0630
5	-0.0051
6	-0.1761
7	-0.1689
8	-0.2130
9	-0.0832
10	-0.0217
11	-0.0366
12	0.0454
13	-0.0429
14	-0.0335
15	0.0460
16	0.0901
17	0.0545
18	0.1053
19	-0.0209
20	0.0040
21	-0.0134
22	-0.0374
23	0.0977
24	0.0266
25	0.0444
26	0.0418
27	-0.0821
28	0.0368
29	0.1641
30	-0.0201
31	-0.0003
32	-0.0323
33	0.0426
34	-0.0330
35	0.0582
36	0.0145
37	-0.0366
38	0.0547
39	-0.0334
40	0.0122
41	0.0306
42	0.0002
43	-0.0220
44	-0.0310
45	-0.0051
46	-0.0611
47	-0.0650
48	0.1568
49	-0.0077
50	0.0688
51	-0.0335
52	0.0147
53	0.0190
54	-0.0216

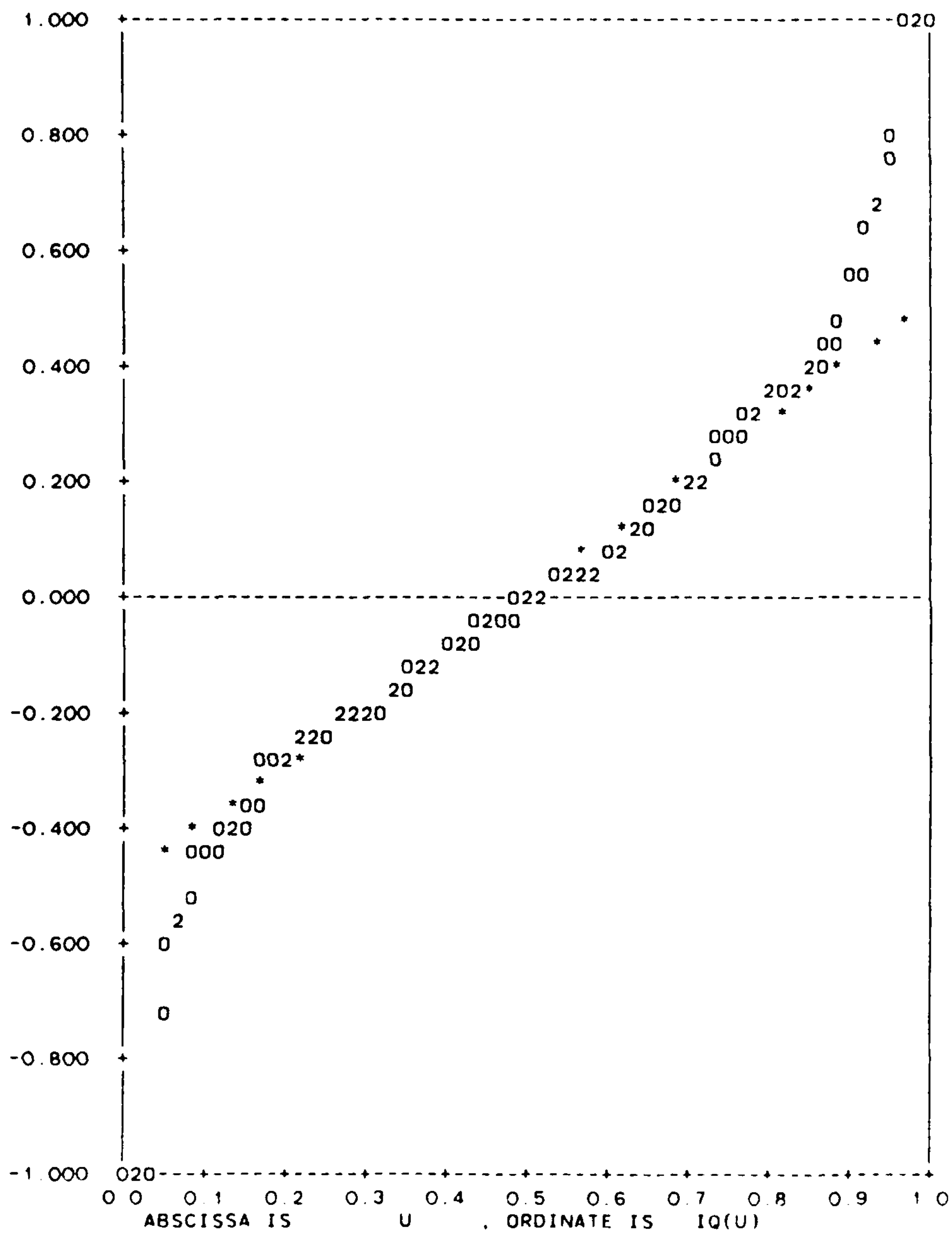


55	-0.0170
56	0.0402
57	-0.1082
58	-0.0330
59	0.1332
60	-0.0127
61	-0.0677
62	0.0884
63	-0.0392
64	-0.0547
65	0.0367
66	0.0321
67	-0.0139
68	0.0000
69	-0.0013
70	-0.0659
71	0.0491
72	0.0242
73	0.0089
74	0.0228
75	-0.0589
76	-0.0286
77	-0.0153
78	-0.0025
79	-0.0150
80	0.0134
81	-0.0776
82	0.0041
83	-0.0330
84	-0.0178
85	-0.0877
86	-0.0048
87	-0.0071
88	0.0056
89	-0.0583
90	-0.0411
91	0.0337
92	0.0000
93	0.0829
94	-0.0290
95	0.0470
96	0.0229
97	-0.0457
98	-0.0115
99	0.0511
100	0.0114
101	-0.0452
102	0.0010
103	-0.0634
104	-0.0449
105	-0.0104
106	0.0145



C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER

INFORMATIVE QUANTILE - PARTIAL AUTOCORRELATIONS



U	0.01000	0.05000	0.10000	0.25000	0.75000	0.90000	0.93000	0.99000
IQ(U)	-5.59618	-0.68707	-0.43670	-0.22054	0.27946	0.54405	0.73669	4.65798

FULLY NON-PARAMETRIC ANALYSIS

C WOLFERS SUNSPOT NUMBERS, 1749-1963 WOLFER
PARTIAL AUTOCORRELATIONS

DESCRIPTIVE STATISTICS

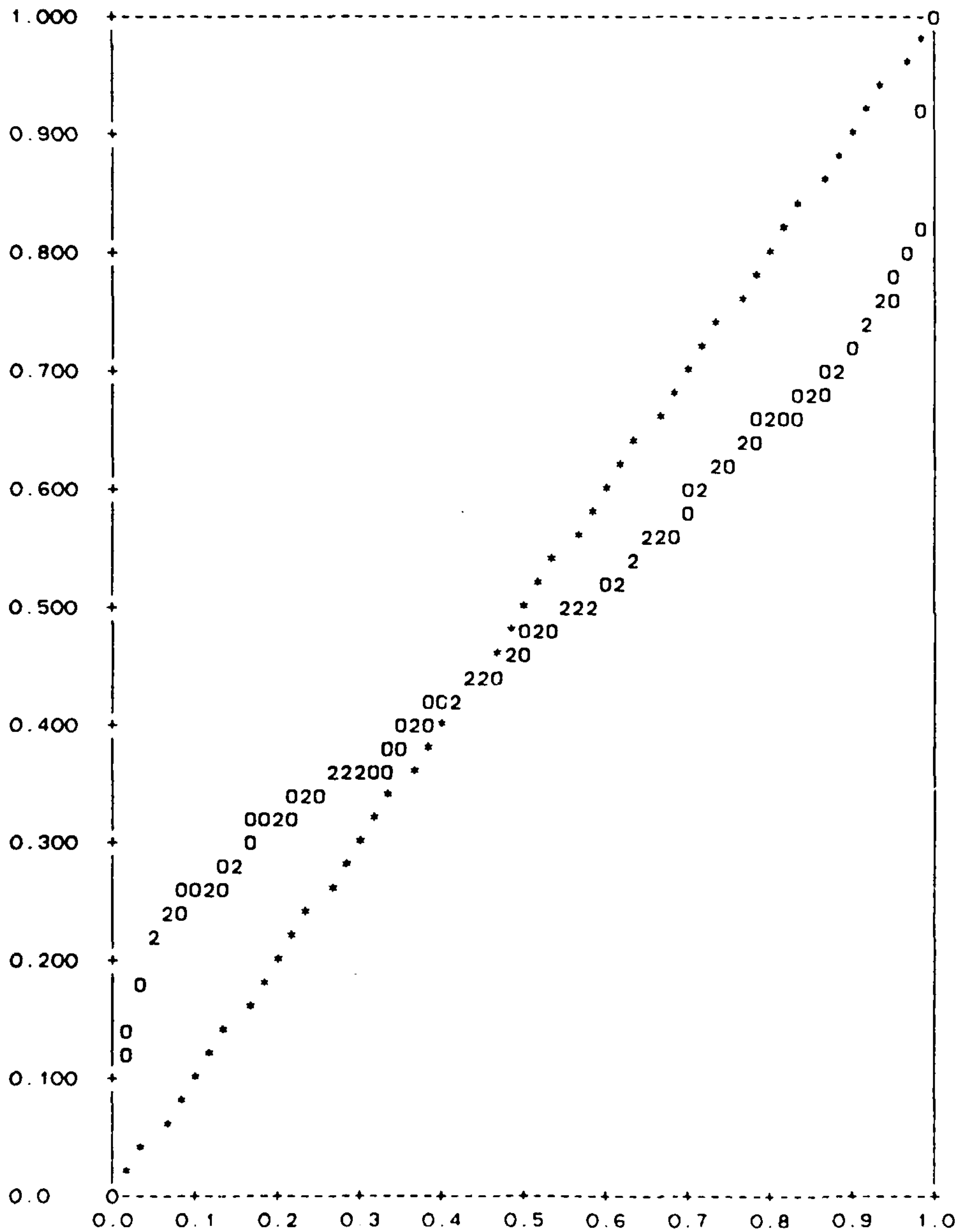
SAMPLE SIZE	LOWER QUARTILE	MEDIAN	UPPER QUARTILE	INT_QUARTL RANGE
106	-.3658E-01	-.6100E-02	.3252E-01	.6910E-01

SUMSQ/N	MEAN	VARIANCE	STD DEV	MEAN IQ	STD DEV IQ	LOG STD IQ
.1413E-01	-.6028E-02	.1423E-01	.1193	.5221E-03	.8631	-.1472

AV. LOG SPACINGS	AV. LOG W. SPACINGS	AV. LOG HYP. FQ	SIGMA ZERO	LOG SIGMA ZERO
.33825362	-.62412298	-1.3855381	.65497315	-.42316103

RAW DISTRIBUTION D(U) NORMAL CASE

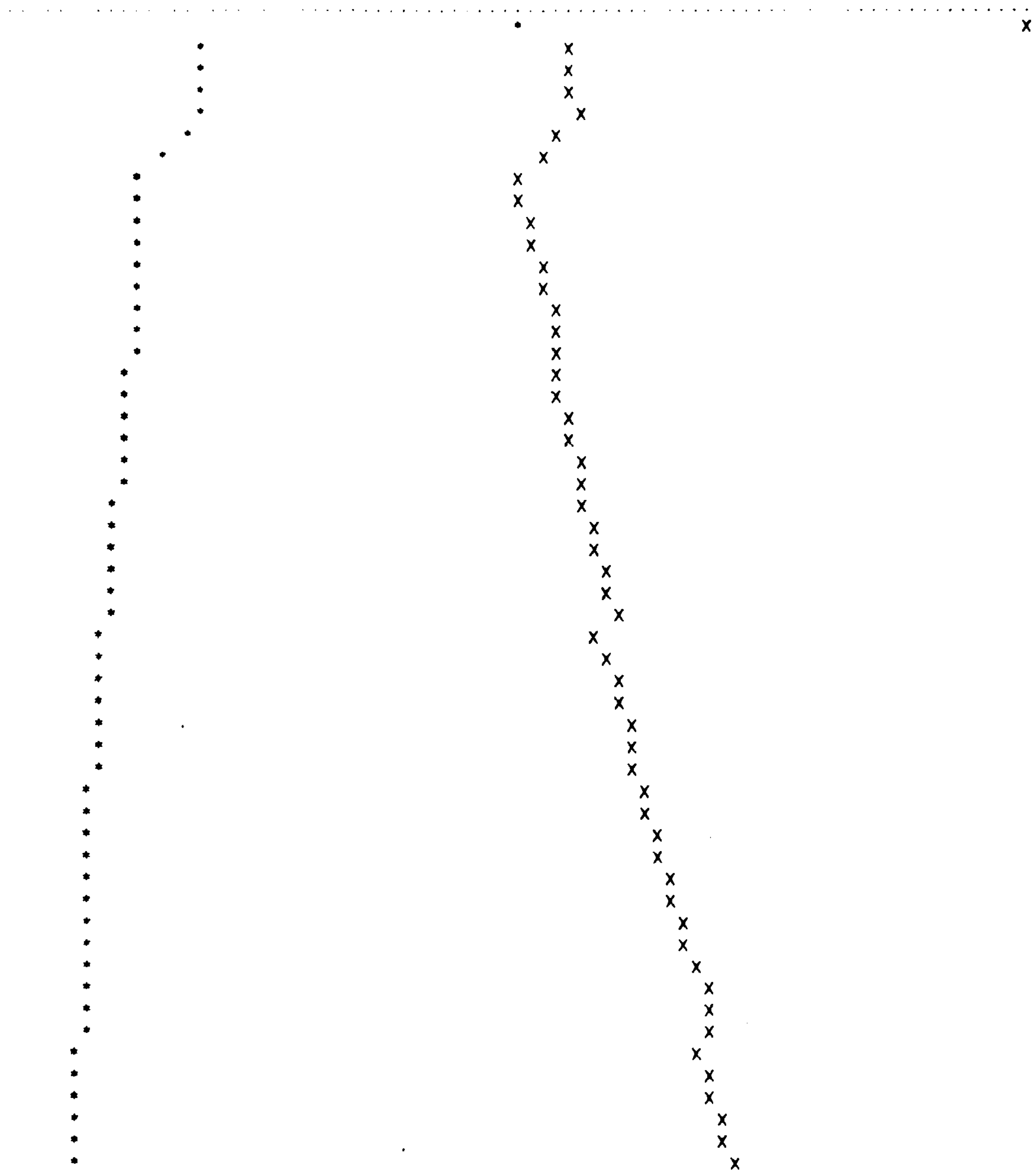
D+ = 1.8035 AT U = 0.0519 . D- = -1.9283 AT U = 0.8821



ABSCISSA IS U , ORDINATE IS D(U)

ORDER BY AIC = 8

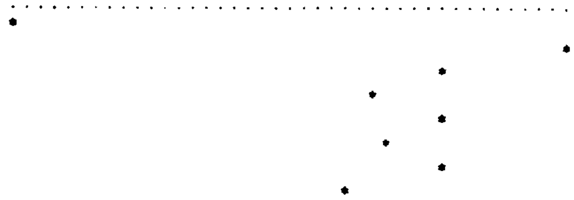
I	ALRV	AIC
1	-1 1267	-1.1174
2	-1.7306	-1.7120
3	-1.7355	-1.7076
4	-1.7395	-1.7023
5	-1.7395	-1.6930
6	-1.7710	-1.7152
7	-1.8000	-1.7348
8	-1.8464	-1.7719
9	-1.8533	-1.7696
10	-1.8538	-1.7608
11	-1.8551	-1.7528
12	-1.8572	-1.7456
13	-1.8590	-1.7381
14	-1.8602	-1.7299
15	-1.8623	-1.7227
16	-1.8704	-1.7216
17	-1.8734	-1.7152
18	-1.8845	-1.7171
19	-1.8850	-1.7082
20	-1.8850	-1.6989
21	-1.8852	-1.6898
22	-1.8866	-1.6819
23	-1.8962	-1.6822
24	-1.8969	-1.6736
25	-1.8988	-1.6663
26	-1.9006	-1.6587
27	-1.9073	-1.6562
28	-1.9087	-1.6482
29	-1.9360	-1.6662
30	-1.9364	-1.6573
31	-1.9364	-1.6480
32	-1.9374	-1.6398
33	-1.9393	-1.6323
34	-1.9404	-1.6241
35	-1.9437	-1.6182
36	-1.9440	-1.6091
37	-1.9453	-1.6011
38	-1.9483	-1.5948
39	-1.9494	-1.5866
40	-1.9496	-1.5775
41	-1.9505	-1.5691
42	-1.9505	-1.5598
43	-1.9510	-1.5510
44	-1.9519	-1.5426
45	-1.9520	-1.5334
46	-1.9557	-1.5278
47	-1.9600	-1.5227
48	-1.9[illegible]9	-1.5384
49	-1.[illegible]9	-1.5291
50	-1.9897	-1.5245
51	-1.9908	-1.5164
52	-1.9910	-1.5073
53	-1.9914	-1.4983



54	-1.9918	-1.4895
55	-1.9921	-1.4805
56	-1.9937	-1.4728
57	-2.0055	-1.4753
58	-2.0066	-1.4671
59	-2.0245	-1.4757
60	-2.0247	-1.4665
61	-2.0293	-1.4618
62	-2.0371	-1.4604
63	-2.0387	-1.4526
64	-2.0417	-1.4463
65	-2.0430	-1.4384
66	-2.0440	-1.4301
67	-2.0442	-1.4210
68	-2.0442	-1.4117
69	-2.0442	-1.4024
70	-2.0486	-1.3974
71	-2.0510	-1.3905
72	-2.0516	-1.3818
73	-2.0517	-1.3726
74	-2.0522	-1.3638
75	-2.0557	-1.3580
76	-2.0565	-1.3495
77	-2.0567	-1.3404
78	-2.0567	-1.3312
79	-2.0570	-1.3221
80	-2.0571	-1.3130
81	-2.0632	-1.3097
82	-2.0632	-1.3004
83	-2.0643	-1.2922
84	-2.0646	-1.2832
85	-2.0723	-1.2816
86	-2.0723	-1.2723
87	-2.0724	-1.2631
88	-2.0724	-1.2538
89	-2.0758	-1.2479
90	-2.0775	-1.2403
91	-2.0786	-1.2321
92	-2.0786	-1.2228
93	-2.0855	-1.2204
94	-2.0864	-1.2120
95	-2.0886	-1.2049
96	-2.0891	-1.1961
97	-2.0912	-1.1889
98	-2.0913	-1.1797
99	-2.0940	-1.1730
100	-2.0941	-1.1638
101	-2.0961	-1.1566
102	-2.0961	-1.1473
103	-2.1002	-1.1420
104	-2.1022	-1.1347
105	-2.1023	-1.1255
106	-2.1025	-1.1164

```
MININ( 1) =  2  VALUE   -5.5513239
MININ( 2) = 18  VALUE   -5.5509377
MININ( 3) = 29  VALUE   -5.2114515
MININ( 4) = 48  VALUE   -4.3608418
```

```
MININ( 5) =  59  VALUE        -3.9200706

MIN1 =   8  CAT(MIN1) =       -5.9006128
MIN2 =   2  CAT(MIN2) =       -5.5513239

FOR ORDER  8 AR MODEL, RVAR = 0.1578100


COEFFICIENTS FOR BEST     ORDER :

    I       ALPH
--------------
    1    -1.2661
    2     0.5007
    3     0.1208
    4    -0.1356
    5     0.1048
    6    -0.0600
    7     0.1084
    8    -0.2130
```

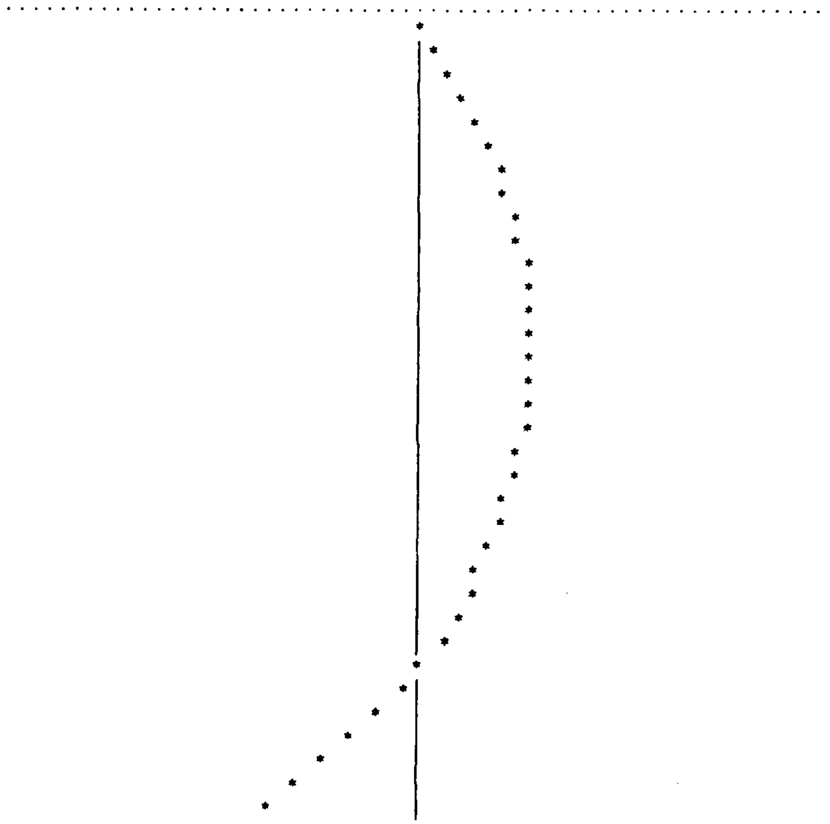


DELTA MEMORY FUNCTION

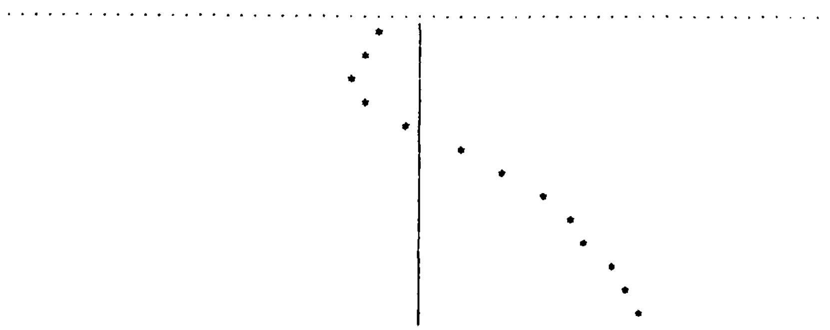
BEST ORDER AR SPECTRAL DENSITY

PLOT 1 - LAG 1 IS AT FREQUENCY 0
PLOT 2 - LAG 1 IS AT FREQUENCY 0.09091

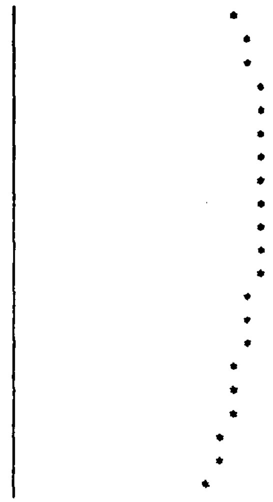
I	DELT
1	0.0290
2	0.0960
3	0.1847
4	0.2828
5	0.3805
6	0.4716
7	0.5529
8	0.6230
9	0.6816
10	0.7290
11	0.7657
12	0.7924
13	0.8097
14	0.8180
15	0.8178
16	0.8094
17	0.7930
18	0.7687
19	0.7365
20	0.6963
21	0.6480
22	0.5910
23	0.5250
24	0.4494
25	0.3634
26	0.2658
27	0.1556
28	0.0310
29	-0.1099
30	-0.2694
31	-0.4504
32	-0.6557
33	-0.8876
34	-1.1450



I	DELT
1	-0.3046
2	-0.4414
3	-0.4868
4	-0.3618
5	-0.0775
6	0.2653
7	0.5851
8	0.8536
9	1.0700
10	1.2421
11	1.3785
12	1.4866
13	1.5721

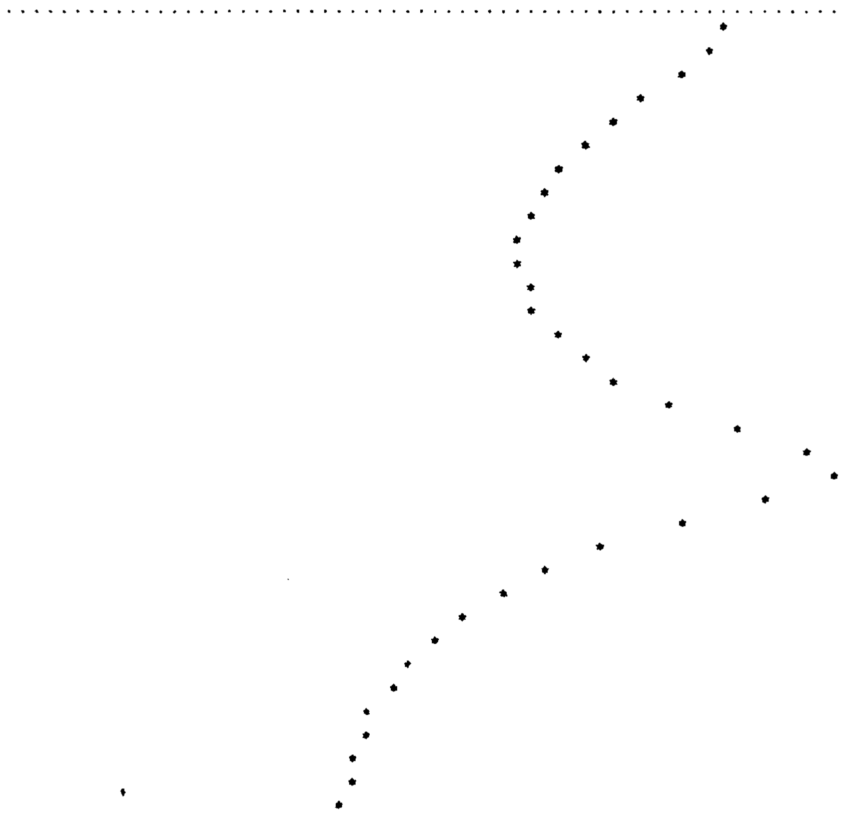


14	1.6394
15	1.6919
16	1.7319
17	1.7615
18	1.7823
19	1.7953
20	1.8015
21	1.8017
22	1.7965
23	1.7863
24	1.7716
25	1.7527
26	1.7300
27	1.7037
28	1.6740
29	1.6414
30	1.6059
31	1.5680
32	1.5279
33	1.4861
34	1.4429

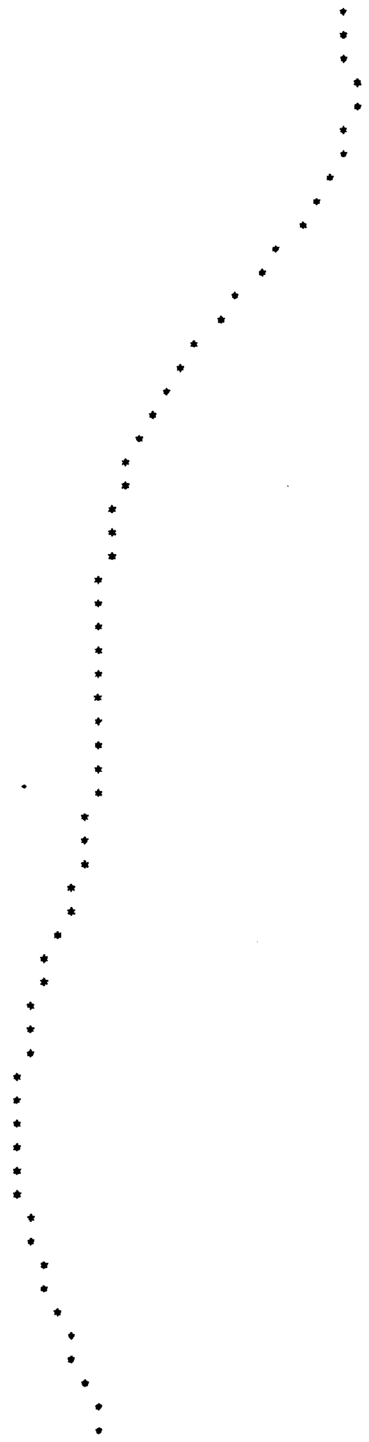


SPECTRA, SPLMIN = -4.01750 SPLMAX = 2.65434 SPLR = 6.67184

FREQUENCY	PERIOD	SPEC
0.0	0.0	6.16723
0.00521	192.00005	5.52201
0.01042	96.00002	4.23677
0.01562	64.00002	3.11116
0.02083	48.00000	2.33295
0.02604	38.40001	1.82902
0.03125	32.00000	1.50718
0.03646	27.42857	1.30455
0.04167	24.00000	1.18389
0.04687	21.33333	1.12505
0.05208	19.20000	1.11954
0.05729	17.45454	1.16844
0.06250	16.00000	1.28359
0.06771	14.76923	1.49375
0.07292	13.71429	1.86032
0.07812	12.80001	2.51875
0.08333	12.00000	3.79173
0.08854	11.29412	6.47598
0.09375	10.66667	11.72606
0.09896	10.10526	14.16819
0.10417	9.60000	8.24191
0.10937	9.14286	4.09531
0.11458	8.72727	2.26946
0.11979	8.34783	1.41509
0.12500	8.00000	0.96823
0.13021	7.68000	0.71193
0.13542	7.38462	0.55439
0.14062	7.11111	0.45262
0.14583	6.85715	0.38468
0.15104	6.62069	0.33858
0.15625	6.40000	0.30739
0.16146	6.19355	0.28690
0.16667	6.00000	0.27446
0.17187	5.81819	0.26827



0.17708	5.64706	0.26692
0.18229	5.48572	0.26905
0.18750	5.33334	0.27298
0.19271	5.18919	0.27646
0.19792	5.05263	0.27668
0.20312	4.92308	0.27074
0.20833	4.80000	0.25683
0.21354	4.68293	0.23525
0.21875	4.57143	0.20851
0.22396	4.46512	0.18011
0.22917	4.36364	0.15309
0.23437	4.26667	0.12924
0.23958	4.17392	0.10920
0.24479	4.08511	0.09287
0.25000	4.00000	0.07979
0.25521	3.91837	0.06941
0.26042	3.84000	0.06121
0.26562	3.76471	0.05475
0.27083	3.69231	0.04968
0.27604	3.62264	0.04572
0.28125	3.55556	0.04266
0.28646	3.49091	0.04032
0.29167	3.42857	0.03858
0.29687	3.36842	0.03734
0.30208	3.31035	0.03651
0.30729	3.25424	0.03601
0.31250	3.20000	0.03576
0.31771	3.14754	0.03571
0.32292	3.09678	0.03574
0.32812	3.04762	0.03579
0.33333	3.00000	0.03574
0.33854	2.95385	0.03550
0.34375	2.90909	0.03500
0.34896	2.86567	0.03419
0.35417	2.82353	0.03306
0.35937	2.78261	0.03166
0.36458	2.74286	0.03006
0.36979	2.70423	0.02835
0.37500	2.66667	0.02662
0.38021	2.63014	0.02497
0.38542	2.59459	0.02343
0.39062	2.56000	0.02207
0.39583	2.52632	0.02089
0.40104	2.49351	0.01991
0.40625	2.46154	0.01913
0.41146	2.43038	0.01856
0.41667	2.40000	0.01818
0.42187	2.37037	0.01801
0.42708	2.34146	0.01804
0.43229	2.31325	0.01828
0.43750	2.28571	0.01873
0.44271	2.25882	0.01943
0.44792	2.23256	0.02038
0.45312	2.20690	0.02161
0.45833	2.18182	0.02313
0.46354	2.15730	0.02497
0.46875	2.13333	0.02710
0.47396	2.10989	0.02949
0.47917	2.08696	0.03200
0.48437	2.06452	0.03445
0.48958	2.04255	0.03656



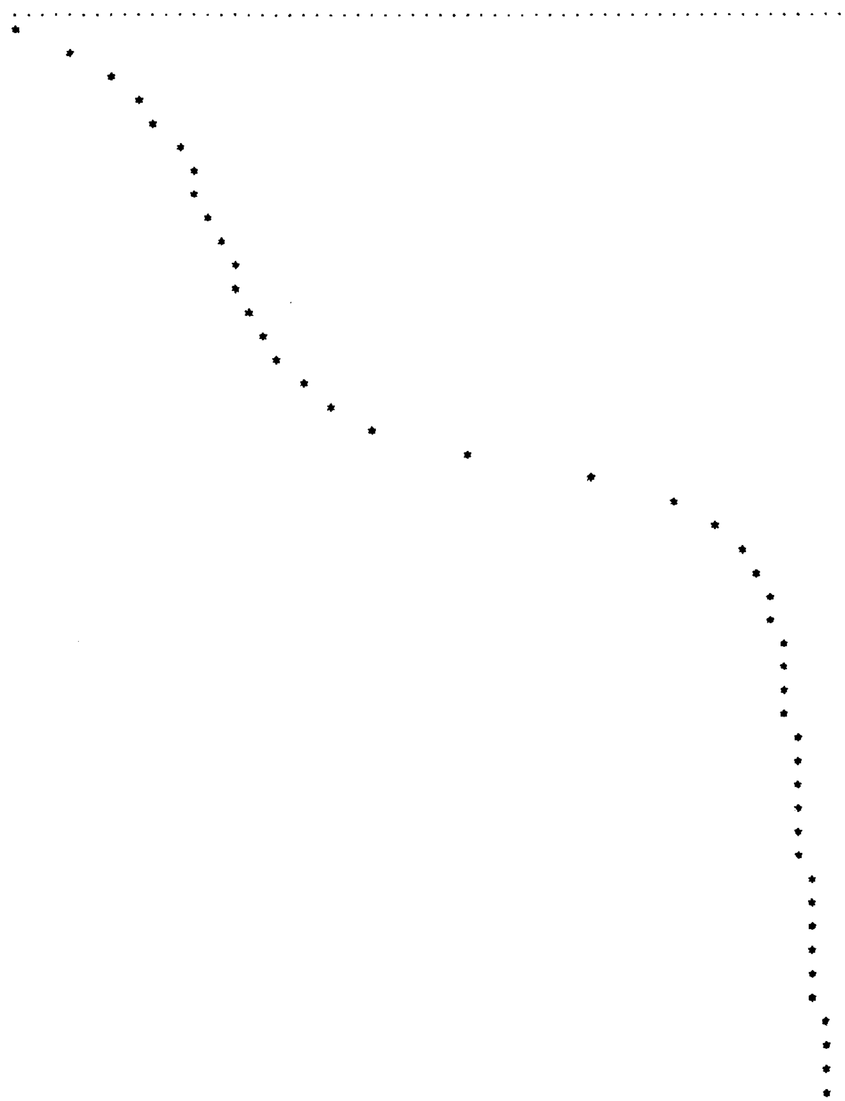
0.49479	2.02105	0.03799
0.50000	2.00000	0.03851

ANALYSIS OF AR SPECTRAL MAXIMA :

I	START	FINAL	NUMIT	SPEC(I)	IER
1	10.3783779	10.2431250	3	88.2042999	1
2	5.1200008	5.1105089	3	1.6746912	1

CUMULATIVE BEST ORDER SPECTRA :

FREQUENCY	PERIOD	SPEC
0.0	0.0	0.03161
0.00521	192.00005	0.09062
0.01042	96.00002	0.13742
0.01562	64.00002	0.17198
0.02083	48.00000	0.19768
0.02604	38.40001	0.21758
0.03125	32.00000	0.23376
0.03646	27.42857	0.24760
0.04167	24.00000	0.26000
0.04687	21.33333	0.27164
0.05208	19.20000	0.28310
0.05729	17.45454	0.29492
0.06250	16.00000	0.30773
0.06771	14.76923	0.32243
0.07292	13.71429	0.34043
0.07812	12.80001	0.36431
0.08333	12.00000	0.39935
0.08854	11.29412	0.45754
0.09375	10.66667	0.56266
0.09896	10.10526	0.70814
0.10417	9.60000	0.80913
0.10937	9.14286	0.85966
0.11458	8.72727	0.88667
0.11979	8.34783	0.90299
0.12500	8.00000	0.91389
0.13021	7.68000	0.92177
0.13542	7.38462	0.92781
0.14062	7.11111	0.93268
0.14583	6.85715	0.93678
0.15104	6.62069	0.94036
0.15625	6.40000	0.94358
0.16146	6.19355	0.94657
0.16667	6.00000	0.94941
0.17187	5.81819	0.95217
0.17708	5.64706	0.95491
0.18229	5.48572	0.95766
0.18750	5.33334	0.96045
0.19271	5.18919	0.96327
0.19792	5.05263	0.96611
0.20312	4.92308	0.96891
0.20833	4.80000	0.97158
0.21354	4.68293	0.97405
0.21875	4.57143	0.97626
0.22396	4.46512	0.97818
0.22917	4.36364	0.97982
0.23437	4.26667	0.98120



0.23958	4.17392	0.98237	*
0.24479	4.08511	0.98336	*
0.25000	4.00000	0.98421	*
0.25521	3.91837	0.98495	*
0.26042	3.84000	0.98559	*
0.26562	3.76471	0.98617	*
0.27083	3.69231	0.98669	*
0.27604	3.62264	0.98717	*
0.28125	3.55556	0.98761	*
0.28646	3.49091	0.98803	*
0.29167	3.42857	0.98843	*
0.29687	3.36842	0.98882	*
0.30208	3.31035	0.98919	*
0.30729	3.25424	0.98956	*
0.31250	3.20000	0.98993	*
0.31771	3.14754	0.99030	*
0.32292	3.09678	0.99066	*
0.32812	3.04762	0.99103	*
0.33333	3.00000	0.99140	*
0.33854	2.95385	0.99176	*
0.34375	2.90909	0.99212	*
0.34896	2.86567	0.99247	*
0.35417	2.82353	0.99282	*
0.35937	2.78261	0.99314	*
0.36458	2.74286	0.99346	*
0.36979	2.70423	0.99375	*
0.37500	2.66667	0.99403	*
0.38021	2.63014	0.99429	*
0.38542	2.59459	0.99453	*
0.39062	2.56000	0.99476	*
0.39583	2.52632	0.99498	*
0.40104	2.49351	0.99519	*
0.40625	2.46154	0.99538	*
0.41146	2.43038	0.99557	*
0.41667	2.40000	0.99576	*
0.42187	2.37037	0.99595	*
0.42708	2.34146	0.99613	*
0.43229	2.31325	0.99632	*
0.43750	2.28571	0.99651	*
0.44271	2.25882	0.99671	*
0.44792	2.23256	0.99691	*
0.45312	2.20690	0.99713	*
0.45833	2.18182	0.99736	*
0.46354	2.15730	0.99761	*
0.46875	2.13333	0.99789	*
0.47396	2.10989	0.99818	*
0.47917	2.08696	0.99850	*
0.48437	2.06452	0.99885	*
0.48958	2.04255	0.99922	*
0.49479	2.02105	0.99961	*
0.50000	2.00000	1.00000	*

FOR ORDER 2 AR MODEL, RVAR = 0.1771755

COEFFICIENTS FOR 2ND BEST ORDER :

I	ALPH	
1	-1.3757	*
2	0.6733	*

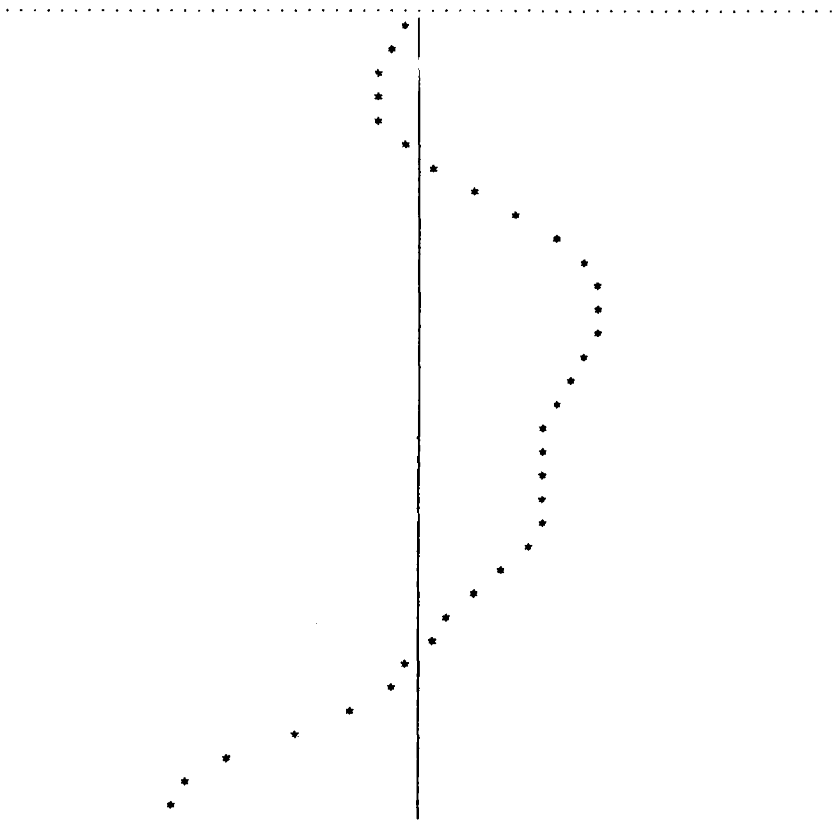
TRUNCATION POINT FOR SMOOTHED PERIODOGRAM = 106

DELTA MEMORY FUNCTION

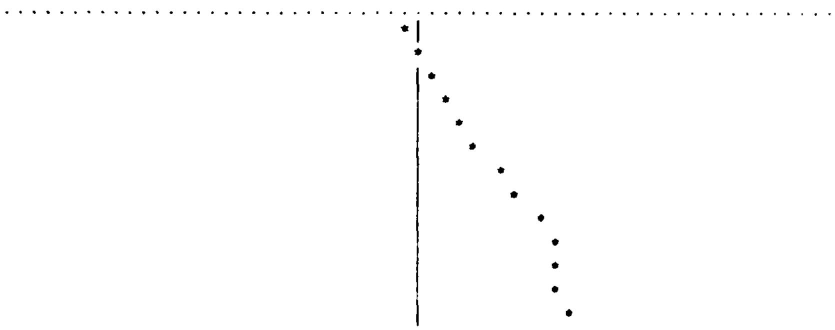
SMOOTHED PERIODOGRAM - PARZEN WINDOW

PLOT 1 - LAG 1 IS AT FREQUENCY 0
PLOT 2 - LAG 1 IS AT FREQUENCY 0.09091

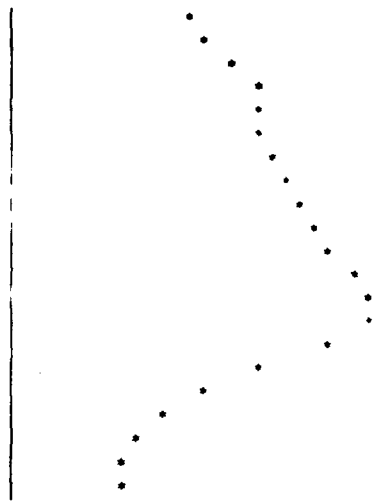
I	DELT
1	-0.0646
2	-0.1891
3	-0.2939
4	-0.3258
5	-0.2612
6	-0.1012
7	0.1355
8	0.4183
9	0.7136
10	0.9893
11	1.2060
12	1.3207
13	1.3335
14	1.2931
15	1.2284
16	1.1333
17	1.0212
18	0.9361
19	0.9076
20	0.9236
21	0.9404
22	0.9093
23	0.8034
24	0.6315
25	0.4279
26	0.2311
27	0.0677
28	-0.0641
29	-0.2254
30	-0.5184
31	-0.9424
32	-1.3645
33	-1.6730
34	-1.8278



I	DELT
1	-0.0588
2	0.0279
3	0.1400
4	0.2366
5	0.3146
6	0.4105
7	0.5550
8	0.7358
9	0.8972
10	0.9854
11	1.0113
12	1.0364
13	1.1117



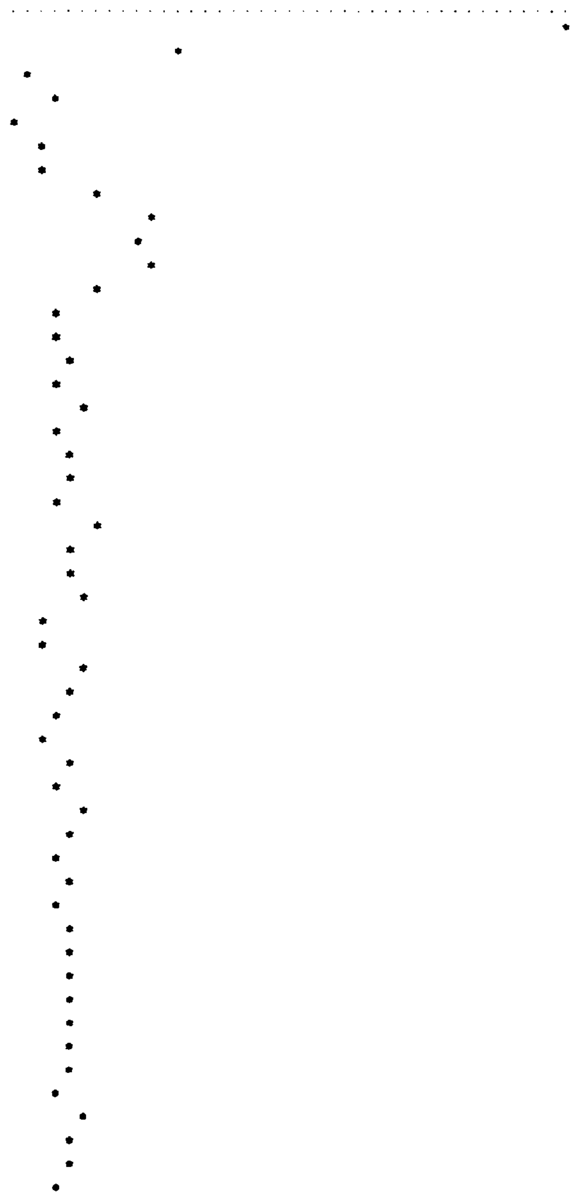
14	1.2528
15	1.4421
16	1.6297
17	1.7569
18	1.8125
19	1.8472
20	1.9112
21	2.0112
22	2.1210
23	2.2209
24	2.3272
25	2.4687
26	2.6194
27	2.6256
28	2.3258
29	1.8447
30	1.3987
31	1.0772
32	0.8867
33	0.7999
34	0.7787



MIXED SCHEME SELECT PROCEDURE

SIGMA INFINITY SQUARED (VIA SMOOTHED PER.) = 0.14973

I	CEPC
1	1.2584
2	0.2894
3	-0.1040
4	-0.0483
5	-0.1374
6	-0.0620
7	-0.0690
8	0.0891
9	0.2144
10	0.1857
11	0.2058
12	0.0863
13	-0.0399
14	-0.0398
15	0.0035
16	-0.0343
17	0.0359
18	-0.0368
19	-0.0147
20	0.0089
21	-0.0206
22	0.0680
23	0.0184
24	-0.0082
25	0.0230
26	-0.0779
27	-0.0738
28	0.0516
29	-0.0089
30	-0.0329
31	-0.0510
32	0.0151
33	-0.0307
34	0.0280
35	0.0035
36	-0.0383
37	0.0049
38	-0.0281
39	-0.0115
40	0.0020
41	0.0015
42	0.0003
43	-0.0099
44	0.0120
45	0.0038
46	-0.0230
47	0.0266
48	-0.0009
49	0.0095
50	-0.0168

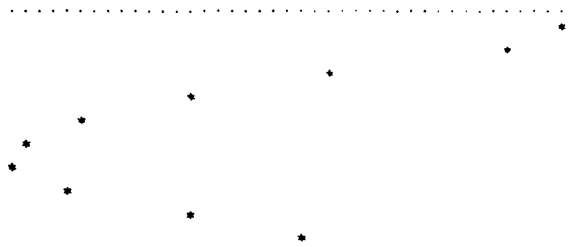


51	-0.0080	*
52	-0.0011	*
53	-0.0027	*
54	-0.0082	*
55	0.0115	*
56	-0.0046	*
57	-0.0098	*
58	0.0143	*
59	0.0067	*
60	-0.0074	*
61	0.0035	*
62	-0.0026	*
63	-0.0040	*
64	-0.0010	*
65	0.0024	*
66	-0.0002	*
67	-0.0003	*
68	0.0007	*
69	-0.0042	*
70	0.0006	*
71	0.0010	*
72	-0.0004	*
73	0.0010	*
74	-0.0001	*
75	-0.0007	*
76	-0.0005	*
77	-0.0006	*
78	0.0005	*
79	0.0003	*
80	-0.0000	*
81	-0.0006	*
82	0.0002	*
83	-0.0001	*
84	-0.0001	*
85	0.0010	*
86	0.0000	*
87	-0.0002	*
88	0.0001	*
89	-0.0002	*
90	-0.0001	*
91	0.0001	*
92	-0.0000	*
93	-0.0001	*
94	0.0000	*
95	-0.0000	*
96	-0.0000	*
97	-0.0000	*
98	-0.0000	*
99	0.0000	*
100	0.0001	*
101	-0.0001	*
102	0.0002	*
103	0.0000	*
104	-0.0002	*
105	0.0001	*
106	0.0001	*

MA MODEL VIA CEPSTRAL CORR., RVAR= 0.15258

FIRST 10 COEFFICIENTS OF INFINITE MA:

I	BETA
1	1.2584
2	1.0812
3	0.5923
4	0.1964
5	-0.1355
6	-0.2812
7	-0.3289
8	-0.1596
9	0.1857
10	0.5089



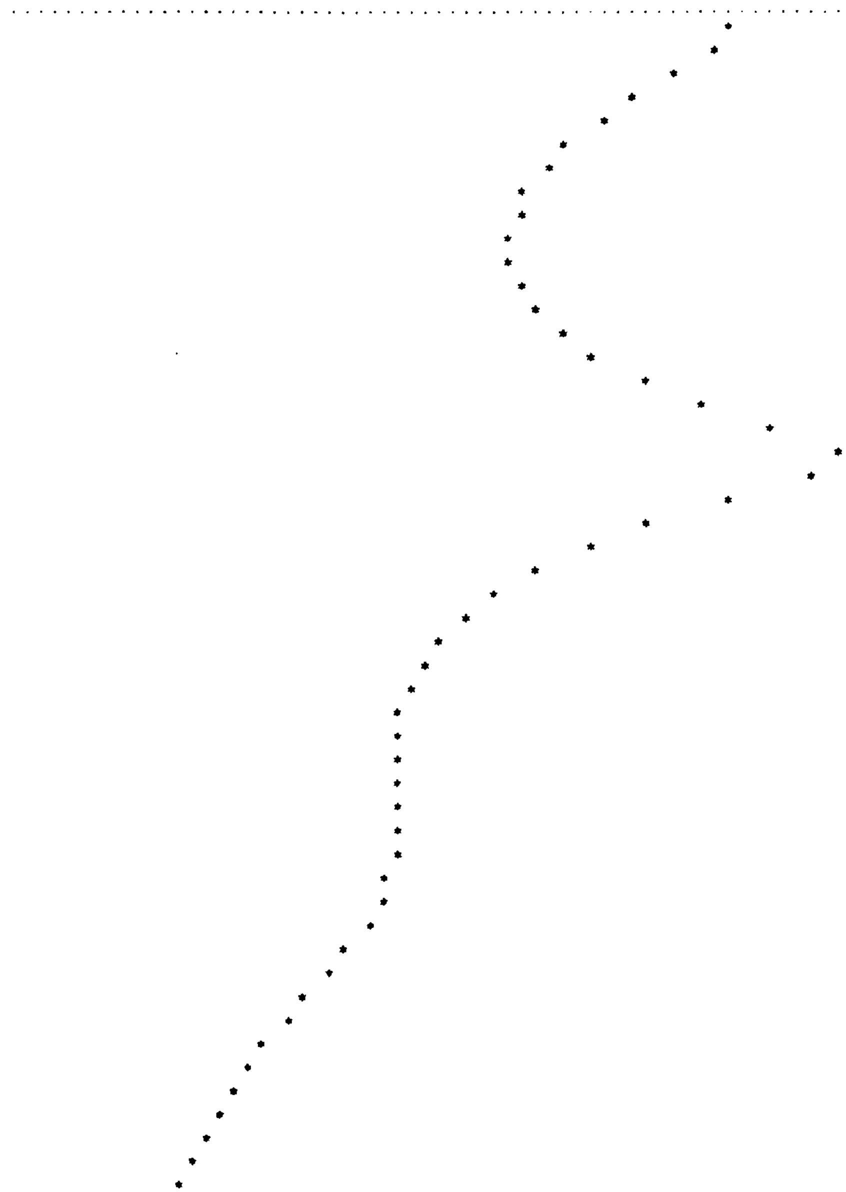
LAG	RYY(V)	REY(V)	RYE(V)	REE(V)	PVH(V)
0	1.0000000	0.1525801	0.1525801	0.1525801	0.1525801
1	0.8212516	0.1920068	0.0	0.0	0.3942013
2	0.4560588	0.1649727	0.0	0.0	0.5725732
3	0.0633344	0.0903800	0.0	0.0	0.6261092
4	-0.2225544	0.0299657	0.0	0.0	0.6319942
5	-0.3457045	-0.0206712	0.0	0.0	0.6347947
6	-0.2882376	-0.0429078	0.0	0.0	0.6468610
7	-0.0899637	-0.0501892	0.0	0.0	0.6633700
8	0.1800877	-0.0243539	0.0	0.0	0.6672572
9	0.4165128	0.0283414	0.0	0.0	0.6725215
10	0.5313844	0.0776423	0.0	0.0	0.7120307
11	0.4940371	0.1126288	0.0	0.0	0.7951690
12	0.3256853	0.1114869	0.0	0.0	0.8766299
13	0.1032266	0.0720053	0.0	0.0	0.9106105
14	-0.0889249	0.0210675	0.0	0.0	0.9135194
15	-0.1988094	-0.0146934	0.0	0.0	0.9149343
16	-0.2163078	-0.0370177	0.0	0.0	0.9239153
17	-0.1530832	-0.0317362	0.0	0.0	0.9305163
18	-0.0489802	-0.0208281	0.0	0.0	0.9333594
19	0.0654125	-0.0017350	0.0	0.0	0.9333791
20	0.1529310	0.0201087	0.0	0.0	0.9360293
21	0.1894133	0.0318503	0.0	0.0	0.9426778
22	0.1709481	0.0423684	0.0	0.0	0.9544426
23	0.0999367	0.0386368	0.0	0.0	0.9642263
24	0.0023017	0.0219768	0.0	0.0	0.9673917
25	-0.0901654	0.0066035	0.0	0.0	0.9676775
26	-0.1519005	-0.0161352	0.0	0.0	0.9693837
27	-0.1613659	-0.0361856	0.0	0.0	0.9779654
28	-0.1248652	-0.0290928	0.0	0.0	0.9835126
29	-0.0760095	-0.0170519	0.0	0.0	0.9854182
30	-0.0262967	-0.0083312	0.0	0.0	0.9858730
31	0.0163787	-0.0058679	0.0	0.0	0.9860987
32	0.0452819	0.0033431	0.0	0.0	0.9861719
33	0.0462332	0.0050808	0.0	0.0	0.9863411
34	0.0221642	0.0104878	0.0	0.0	0.9870619
35	-0.0221239	0.0061283	0.0	0.0	0.9873080
36	-0.0652053	-0.0069687	0.0	0.0	0.9876263
37	-0.0901687	-0.0146585	0.0	0.0	0.9890345
38	-0.0948357	-0.0216443	0.0	0.0	0.9921048
39	-0.0794292	-0.0214378	0.0	0.0	0.9951168
40	-0.0535882	-0.0157604	0.0	0.0	0.9967447
41	-0.0260918	-0.0093130	0.0	0.0	0.9973131
42	-0.0023169	-0.0038292	0.0	0.0	0.9974092
43	0.0139985	-0.0001726	0.0	0.0	0.9974093
44	0.0193393	0.0038699	0.0	0.0	0.9975075
45	0.0129045	0.0058544	0.0	0.0	0.9977320
46	-0.0007422	-0.0001567	0.0	0.0	0.9977322
47	-0.0128917	-0.0022797	0.0	0.0	0.9977662
48	-0.0252214	-0.0060121	0.0	0.0	0.9980031
49	-0.0333354	-0.0063379	0.0	0.0	0.9982663
50	-0.0367247	-0.0084426	0.0	0.0	0.9987334
51	-0.0319753	-0.0080595	0.0	0.0	0.9991591
52	-0.0215767	-0.0060752	0.0	0.0	0.9994010
53	-0.0086154	-0.0028588	0.0	0.0	0.9994545
54	0.0034824	-0.0013836	0.0	0.0	0.9994670
55	0.0132193	0.0007973	0.0	0.0	0.9994711
56	0.0176550	0.0013502	0.0	0.0	0.9994960
57	0.0165059	0.0015813	0.0	0.0	0.9995124
58	0.0104697	0.0021768	0.0	0.0	0.9995434

```
 59     0.0000932     0.0023675    0.0        0.0          0.9995801
 60    -0.0105894    -0.0000931    0.0        0.0          0.9995801
 61    -0.0163468    -0.0024267    0.0        0.0          0.9996187
 62    -0.0159941    -0.0030669    0.0        0.0          0.9996803
 63    -0.0108299    -0.0033139    0.0        0.0          0.9997523
 64    -0.0024386    -0.0021298    0.0        0.0          0.9997820
 65     0.0058578    -0.0005836    0.0        0.0          0.9997842
 66     0.0117297     0.0010471    0.0        0.0          0.9997913
 67     0.0136500     0.0023796    0.0        0.0          0.9998284
 68     0.0115049     0.0026493    0.0        0.0          0.9998744

REGRESSION ESTIMATION STAGEWISE SUMMARY
NUMBER OF VARIABLES IN FULL MODEL =  60
MAXIMUM NUMBER OF COEFFICIENTS =   0

--------------------------------------------------------------------------
VAR IN MODEL        COEFF
   CRIT ADD      CRIT DEL           AIC       FPE ADD       FPE DEL
  19.647263  ************      0.0        145.682007  ************
   VAR ADD      VAR DEL    VAR LAST   NO. PREDS     RES VAR      NO. CYC
        1           0          0          0      1.00000            0
--------------------------------------------------------------------------
VAR IN MODEL        COEFF
          1       0.82125
   CRIT ADD      CRIT DEL           AIC       FPE ADD       FPE DEL
  15.020402    133.641968    -1.117599    97.661102   -445.428711
   VAR ADD      VAR DEL    VAR LAST   NO. PREDS     RES VAR      NO. CYC
        2           1          1          1      0.32555            1
--------------------------------------------------------------------------
VAR IN MODEL        COEFF
          2      -0.67086
          1       1.37219
   CRIT ADD      CRIT DEL           AIC       FPE ADD       FPE DEL
   2.923030     52.665573    -1.710878    18.240295   -176.763809
   VAR ADD      VAR DEL    VAR LAST   NO. PREDS     RES VAR      NO. CYC
        9           2          2          2      0.17903            2
--------------------------------------------------------------------------
VAR IN MODEL        COEFF
          9       0.13737
          2      -0.57049
          1       1.26502
   CRIT ADD      CRIT DEL           AIC       FPE ADD       FPE DEL
   0.678282      5.862629    -1.793608     2.087705    -19.814499
   VAR ADD      VAR DEL    VAR LAST   NO. PREDS     RES VAR      NO. CYC
       19           9          9          3      0.16405            3
--------------------------------------------------------------------------

TO INTERPRET CRITICAL ADD AND DELETE VALUES, NOTE THAT
THEY ARE COMPUTED USING TWICE THE SAMPLE SIZE.

RESIDUAL VARIANCE FROM SUBSET ARMA =  0.1640536

    I       ALPH
---------------   .........................................
    1   -1.2650  *
    2    0.5705                                        *
    3    0.0                               *
    4    0.0                               *
```

5	0.0	
6	0.0	
7	0.0	
8	0.0	
9	-0.1374	

SPECTRA, SPLMIN = -3.98663 SPLMAX = 2.63328 SPLR = 6.61990

FREQUENCY	PERIOD	SPEC
0.0	0.0	5.80647
0.00521	192.00005	5.12338
0.01042	96.00002	3.82384
0.01562	64.00002	2.74827
0.02083	48.00000	2.03749
0.02604	38.40001	1.59216
0.03125	32.00000	1.31561
0.03646	27.42857	1.14723
0.04167	24.00000	1.05317
0.04687	21.33333	1.01635
0.05208	19.20000	1.03122
0.05729	17.45454	1.10237
0.06250	16.00000	1.24722
0.06771	14.76923	1.50526
0.07292	13.71429	1.96232
0.07812	12.80001	2.81611
0.08333	12.00000	4.55856
0.08854	11.29412	8.32127
0.09375	10.66667	13.76791
0.09896	10.10526	11.40815
0.10417	9.60000	5.85731
0.10937	9.14286	3.11324
0.11458	8.72727	1.87876
0.11979	8.34783	1.26359
0.12500	8.00000	0.92453
0.13021	7.68000	0.72302
0.13542	7.38462	0.59697
0.14062	7.11111	0.51585
0.14583	6.85715	0.46346
0.15104	6.62069	0.43059
0.15625	6.40000	0.41158
0.16146	6.19355	0.40253
0.16667	6.00000	0.40011
0.17187	5.81819	0.40079
0.17708	5.64706	0.40030
0.18229	5.48572	0.39382
0.18750	5.33334	0.37729
0.19271	5.18919	0.34931
0.19792	5.05263	0.31222
0.20312	4.92308	0.27096
0.20833	4.80000	0.23059
0.21354	4.68293	0.19451
0.21875	4.57143	0.16412
0.22396	4.46512	0.13947
0.22917	4.36364	0.11991
0.23437	4.26667	0.10457
0.23958	4.17392	0.09262
0.24479	4.08511	0.08338
0.25000	4.00000	0.07626
0.25521	3.91837	0.07084

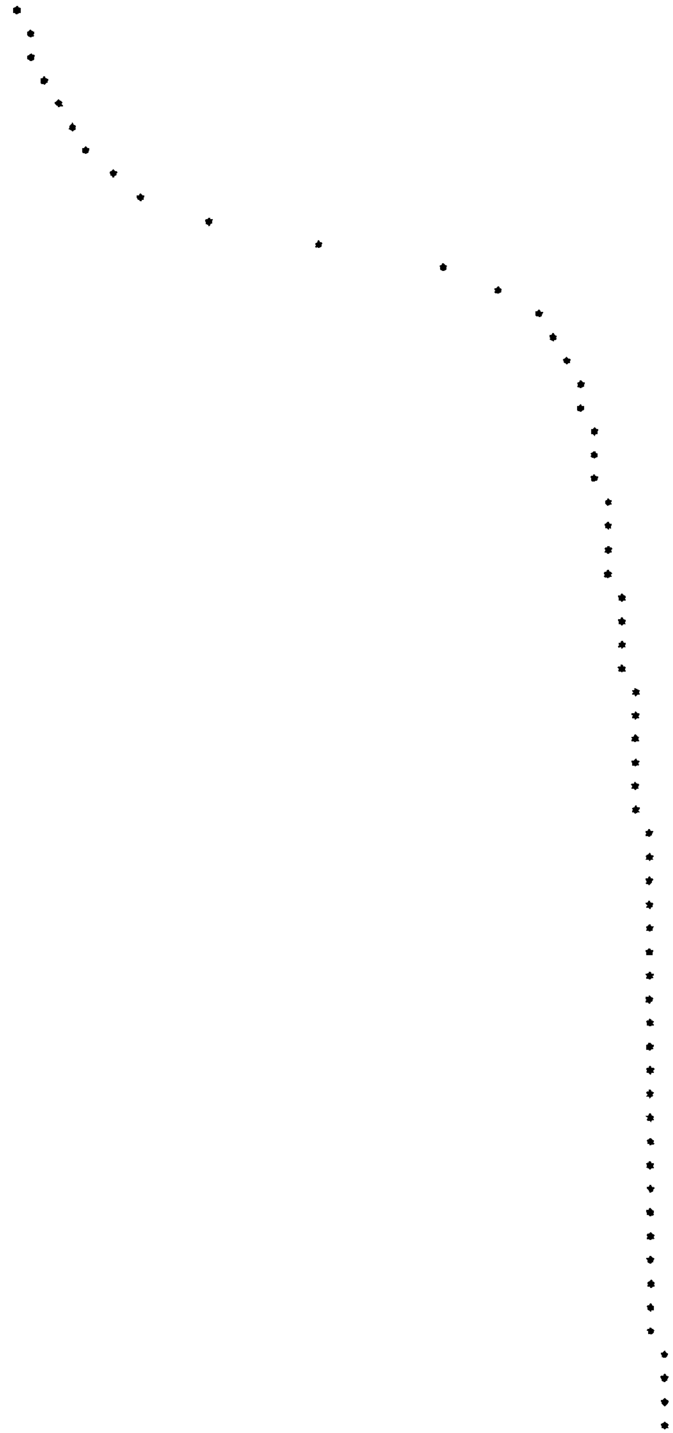


0.26042	3.84000	0.06675
0.26562	3.76471	0.06374
0.27083	3.69231	0.06157
0.27604	3.62264	0.06005
0.28125	3.55556	0.05898
0.28646	3.49091	0.05819
0.29167	3.42857	0.05748
0.29687	3.36842	0.05667
0.30208	3.31035	0.05559
0.30729	3.25424	0.05410
0.31250	3.20000	0.05215
0.31771	3.14754	0.04976
0.32292	3.09678	0.04703
0.32812	3.04762	0.04409
0.33333	3.00000	0.04111
0.33854	2.95385	0.03822
0.34375	2.90909	0.03553
0.34896	2.86567	0.03311
0.35417	2.82353	0.03100
0.35937	2.78261	0.02920
0.36458	2.74286	0.02772
0.36979	2.70423	0.02653
0.37500	2.66667	0.02561
0.38021	2.63014	0.02493
0.38542	2.59459	0.02448
0.39062	2.56000	0.02422
0.39583	2.52632	0.02413
0.40104	2.49351	0.02417
0.40625	2.46154	0.02430
0.41146	2.43038	0.02449
0.41667	2.40000	0.02468
0.42187	2.37037	0.02483
0.42708	2.34146	0.02488
0.43229	2.31325	0.02481
0.43750	2.28571	0.02457
0.44271	2.25882	0.02418
0.44792	2.23256	0.02364
0.45312	2.20690	0.02300
0.45833	2.18182	0.02228
0.46354	2.15730	0.02155
0.46875	2.13333	0.02084
0.47396	2.10989	0.02019
0.47917	2.08696	0.01962
0.48437	2.06452	0.01917
0.48958	2.04255	0.01883
0.49479	2.02105	0.01863
0.50000	2.00000	0.01856

CUMULATIVE SUBSET ARMA SPECTRA :

FREQUENCY	PERIOD	SPEC
0.0	0.0	0.03078
0.00521	192.00005	0.08773
0.01042	96.00002	0.13174
0.01562	64.00002	0.16348
0.02083	48.00000	0.18675
0.02604	38.40001	0.20467
0.03125	32.00000	0.21927
0.03646	27.42857	0.23182

0.04167	24.00000	0.24320
0.04687	21.33333	0.25403
0.05208	19.20000	0.26489
0.05729	17.45454	0.27635
0.06250	16.00000	0.28913
0.06771	14.76923	0.30432
0.07292	13.71429	0.32373
0.07812	12.80001	0.35096
0.08333	12.00000	0.39381
0.08854	11.29412	0.47022
0.09375	10.66667	0.60267
0.09896	10.10526	0.73694
0.10417	9.60000	0.81196
0.10937	9.14286	0.85076
0.11458	8.72727	0.87335
0.11979	8.34783	0.88812
0.12500	8.00000	0.89870
0.13021	7.68000	0.90683
0.13542	7.38462	0.91346
0.14062	7.11111	0.91912
0.14583	6.85715	0.92416
0.15104	6.62069	0.92880
0.15625	6.40000	0.93320
0.16146	6.19355	0.93749
0.16667	6.00000	0.94174
0.17187	5.81819	0.94598
0.17708	5.64706	0.95023
0.18229	5.48572	0.95443
0.18750	5.33334	0.95848
0.19271	5.18919	0.96227
0.19792	5.05263	0.96568
0.20312	4.92308	0.96866
0.20833	4.80000	0.97121
0.21354	4.68293	0.97337
0.21875	4.57143	0.97518
0.22396	4.46512	0.97672
0.22917	4.36364	0.97804
0.23437	4.26667	0.97919
0.23958	4.17392	0.98020
0.24479	4.08511	0.98111
0.25000	4.00000	0.98194
0.25521	3.91837	0.98270
0.26042	3.84000	0.98342
0.26562	3.76471	0.98410
0.27083	3.69231	0.98476
0.27604	3.62264	0.98540
0.28125	3.55556	0.98603
0.28646	3.49091	0.98665
0.29167	3.42857	0.98726
0.29687	3.36842	0.98786
0.30208	3.31035	0.98845
0.30729	3.25424	0.98903
0.31250	3.20000	0.98959
0.31771	3.14754	0.99012
0.32292	3.09678	0.99063
0.32812	3.04762	0.99110
0.33333	3.00000	0.99155
0.33854	2.95385	0.99196
0.34375	2.90909	0.99234
0.34896	2.86567	0.99270
0.35417	2.82353	0.99304



0.35937	2.78261	0.99335
0.36458	2.74286	0.99365
0.36979	2.70423	0.99393
0.37500	2.66667	0.99421
0.38021	2.63014	0.99447
0.38542	2.59459	0.99473
0.39062	2.56000	0.99499
0.39583	2.52632	0.99525
0.40104	2.49351	0.99550
0.40625	2.46154	0.99576
0.41146	2.43038	0.99602
0.41667	2.40000	0.99628
0.42187	2.37037	0.99654
0.42708	2.34146	0.99680
0.43229	2.31325	0.99707
0.43750	2.28571	0.99733
0.44271	2.25882	0.99759
0.44792	2.23256	0.99784
0.45312	2.20690	0.99808
0.45833	2.18182	0.99832
0.46354	2.15730	0.99855
0.46875	2.13333	0.99878
0.47396	2.10989	0.99899
0.47917	2.08696	0.99920
0.48437	2.06452	0.99940
0.48958	2.04255	0.99961
0.49479	2.02105	0.99980
0.50000	2.00000	1.00000

FROM DTARB, RO= 0.14599

PARTIAL AUTOCORRELATIONS VIA BURGS ALGORITHM:

I	PACF
1	-0.8242
2	0.6805
3	0.0780
4	-0.0787
5	0.0115
6	-0.1872
7	-0.1999
8	-0.2494

COEFFICIENTS AND INVERSE CORRELATIONS

SUM OF SQUARES OF COEFFICIENTS = 2.9401

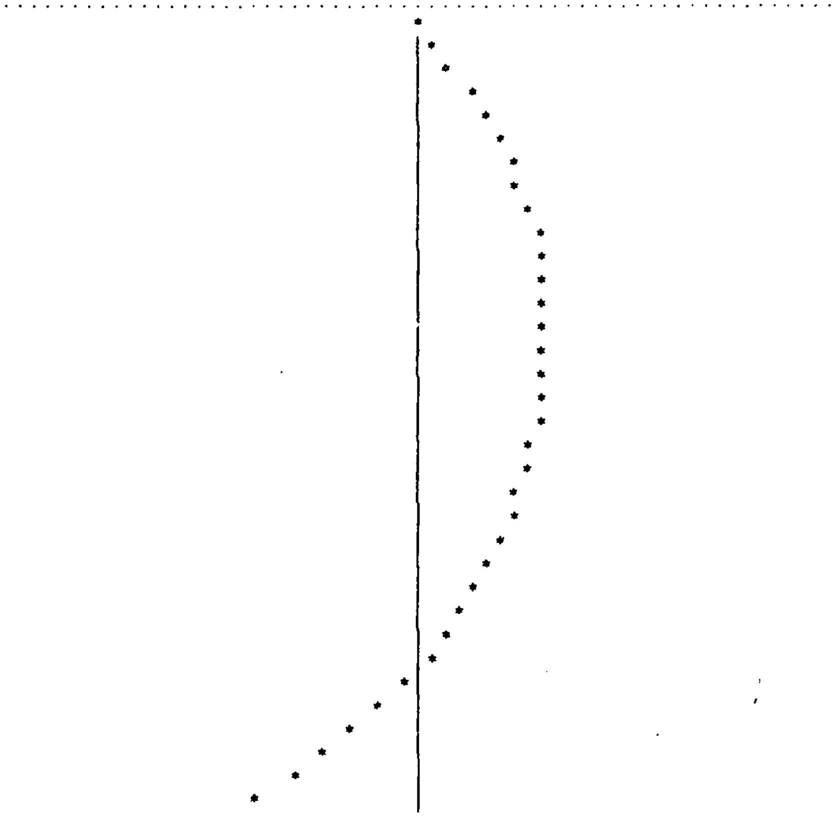
I	ALPH	CORI
1	-1.2539	-0.6327
2	0.4745	0.0845
3	0.1541	0.1227
4	-0.1680	-0.0879
5	0.1048	0.0609
6	-0.0424	-0.1081
7	0.1253	0.1490
8	-0.2494	-0.0848

DELTA MEMORY FUNCTION

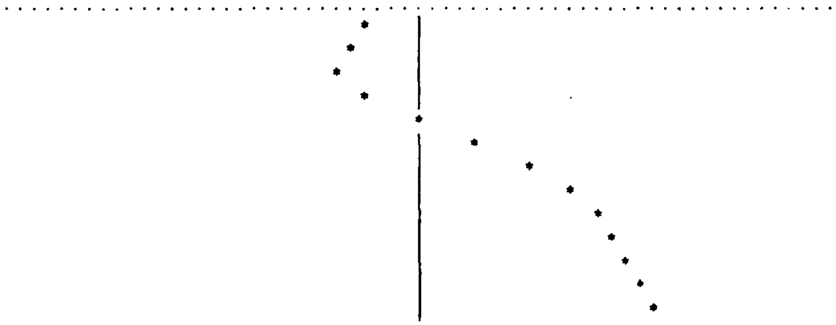
SPECTRAL DENSITY FROM BURG AR MODEL

PLOT 1 - LAG 1 IS AT FREQUENCY 0
PLOT 2 - LAG 1 IS AT FREQUENCY 0.09091

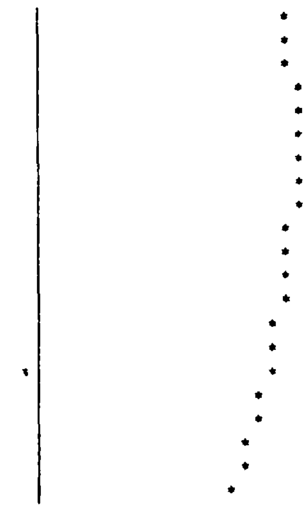
I	DELT
1	0.0402
2	0.1309
3	0.2459
4	0.3669
5	0.4817
6	0.5842
7	0.6721
8	0.7453
9	0.8045
10	0.8508
11	0.8853
12	0.9090
13	0.9228
14	0.9273
15	0.9231
16	0.9105
17	0.8899
18	0.8612
19	0.8246
20	0.7799
21	0.7268
22	0.6650
23	0.5940
24	0.5129
25	0.4209
26	0.3167
27	0.1989
28	0.0656
29	-0.0859
30	-0.2585
31	-0.4562
32	-0.6837
33	-0.9460
34	-1.2460



I	DELT
1	-0.3651
2	-0.5410
3	-0.5890
4	-0.3849
5	0.0115
6	0.4280
7	0.7787
8	1.0529
9	1.2635
10	1.4251
11	1.5494
12	1.6452
13	1.7188

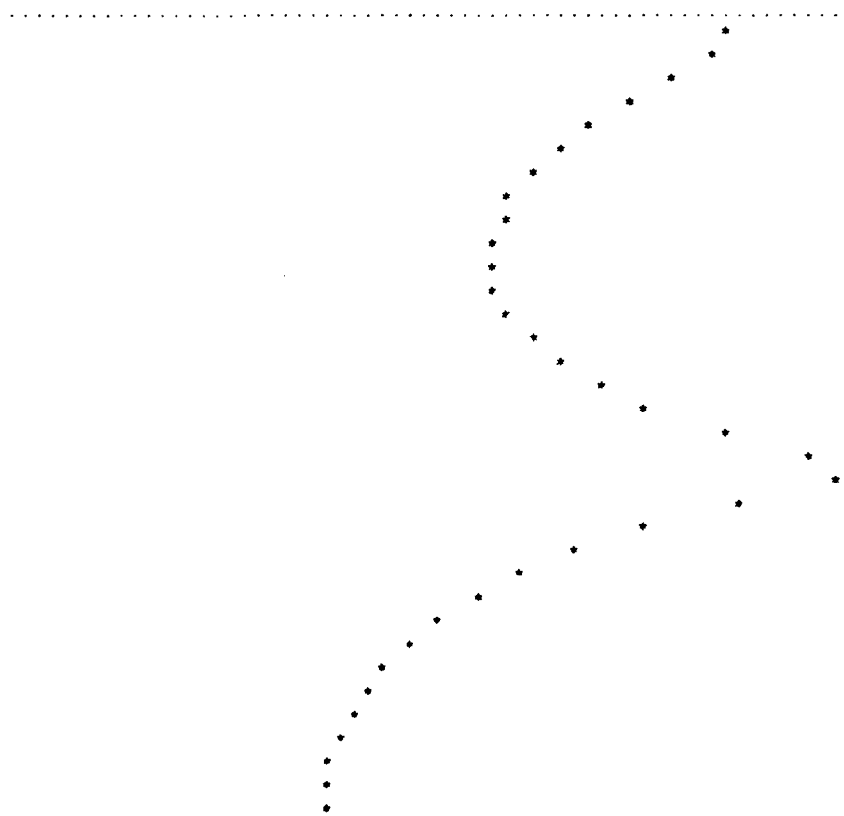


14	1.7749
15	1.8168
16	1.8470
17	1.8674
18	1.8794
19	1.8841
20	1.8823
21	1.8747
22	1.8619
23	1.8442
24	1.8219
25	1.7954
26	1.7649
27	1.7306
28	1.6927
29	1.6513
30	1.6068
31	1.5592
32	1.5088
33	1.4561
34	1.4013

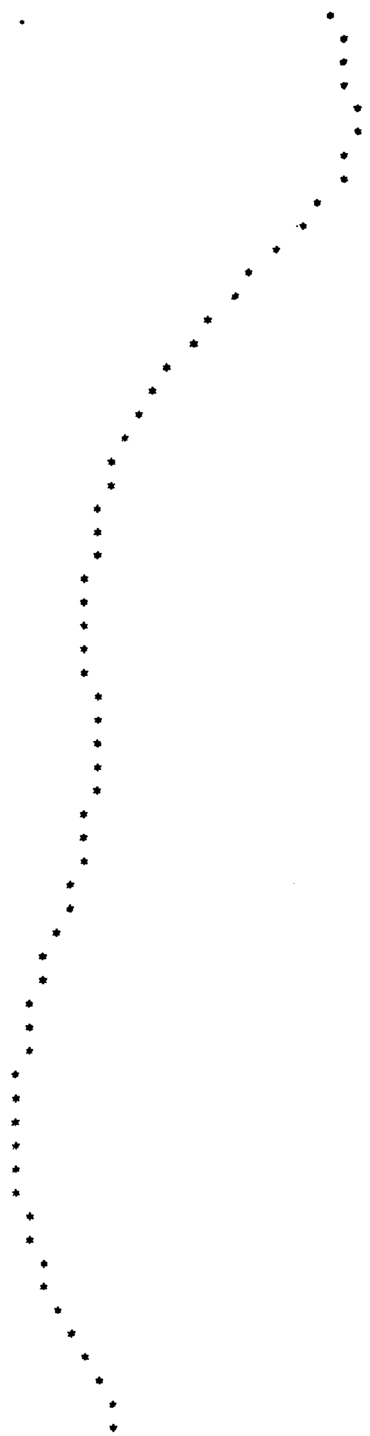


SPECTRA, SPLMIN = -4.07830 SPLMAX = 2.83634 SPLR = 6.91464

FREQUENCY	PERIOD	SPEC
0.0	0.0	7.04114
0.00521	192.00005	6.05410
0.01042	96.00002	4.30453
0.01562	64.00002	2.97043
0.02083	48.00000	2.13851
0.02604	38.40001	1.63452
0.03125	32.00000	1.32580
0.03646	27.42857	1.13644
0.04167	24.00000	1.02550
0.04687	21.33333	0.97188
0.05208	19.20000	0.96680
0.05729	17.45454	1.01098
0.06250	16.00000	1.11562
0.06771	14.76923	1.30849
0.07292	13.71429	1.65071
0.07812	12.80001	2.28379
0.08333	12.00000	3.57398
0.08854	11.29412	6.58283
0.09375	10.66667	13.57119
0.09896	10.10526	16.11069
0.10417	9.60000	7.70292
0.10937	9.14286	3.51911
0.11458	8.72727	1.90166
0.11979	8.34783	1.17818
0.12500	8.00000	0.80647
0.13021	7.68000	0.59507
0.13542	7.38462	0.46587
0.14062	7.11111	0.38293
0.14583	6.85715	0.32814
0.15104	6.62069	0.29167
0.15625	6.40000	0.26792
0.16146	6.19355	0.25359
0.16667	6.00000	0.24672
0.17187	5.81819	0.24607



0.17708	5.64706	0.25078
0.18229	5.48572	0.25993
0.18750	5.33334	0.27208
0.19271	5.18919	0.28465
0.19792	5.05263	0.29348
0.20312	4.92308	0.29336
[illegible]633	4.80000	0.28030
0.21354	4.68293	0.25441
0.21875	4.57143	0.22036
0.22396	4.46512	0.18453
0.22917	4.36364	0.15173
0.23437	4.26667	0.12416
0.23958	4.17392	0.10210
0.24479	4.08511	0.08489
0.25000	4.00000	0.07161
0.25521	3.91837	0.06139
0.26042	3.84000	0.05353
0.26562	3.76471	0.04747
0.27083	3.69231	0.04280
0.27604	3.62264	0.03922
0.28125	3.55556	0.03650
0.28646	3.49091	0.03448
0.29167	3.42857	0.03302
0.29687	3.36842	0.03203
0.30208	3.31035	0.03144
0.30729	3.25424	0.03119
0.31250	3.20000	0.03121
0.31771	3.14754	0.03145
0.32292	3.09678	0.03182
0.32812	3.04762	0.03225
0.33333	3.00000	0.03262
0.33854	2.95385	0.03284
0.34375	2.90909	0.03280
0.34896	2.86567	0.03241
0.35417	2.82353	0.03164
0.35937	2.78261	0.03050
0.36458	2.74286	0.02908
0.36979	2.70423	0.02746
0.37500	2.66667	0.02577
0.38021	2.63014	0.02410
0.38542	2.59459	0.02254
0.39062	2.56000	0.02114
0.39583	2.52632	0.01993
0.40104	2.49351	0.01892
0.40625	2.46154	0.01811
0.41146	2.43038	0.01752
0.41667	2.40000	0.01713
0.42187	2.37037	0.01695
0.42708	2.34146	0.01698
0.43229	2.31325	0.01722
0.43750	2.28571	0.01770
0.44271	2.25882	0.01843
0.44792	2.23256	0.01944
0.45312	2.20690	0.02077
0.45833	2.18182	0.02245
0.46354	2.15730	0.02453
0.46875	2.13333	0.02701
0.47396	2.10989	0.02986
0.47917	2.08696	0.03299
0.48437	2.06452	0.03614
0.48958	2.04255	0.03893



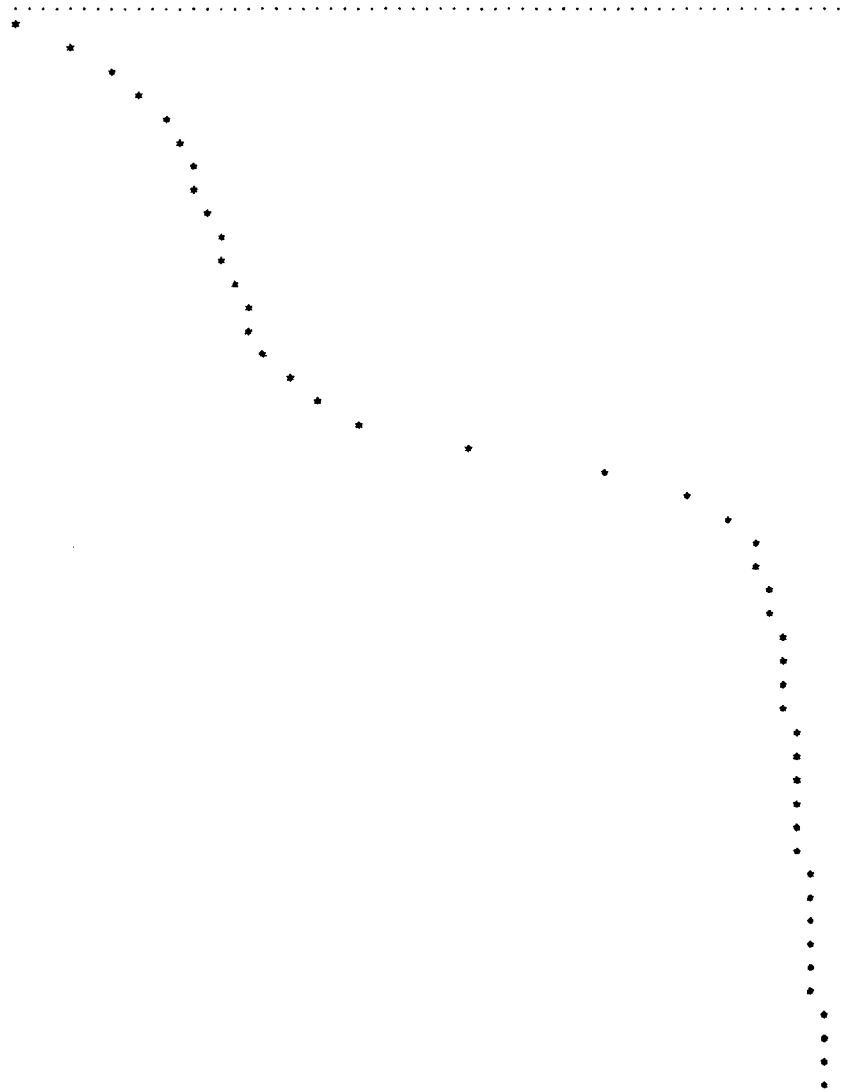
0.49479	2.02105	0.04089
0.50000	2.00000	0.04160

ANALYSIS OF AR SPECTRAL MAXIMA :

I	START	FINAL	NUMIT	SPEC(I)	IER
1	10.3783779	10.2883120	3	126.337372	1
2	4.9870157	4.9862833	2	2.1484814	1

CUMULATIVE SPECTRA FOR BURG AR:

FREQUENCY	PERIOD	SPEC
0.0	0.0	0.03601
0.00521	192.00005	0.10156
0.01042	96.00002	0.14999
0.01562	64.00002	0.18342
0.02083	48.00000	0.20716
0.02604	38.40001	0.22500
0.03125	32.00000	0.23926
0.03646	27.42857	0.25131
0.04167	24.00000	0.26204
0.04687	21.33333	0.27208
0.05208	19.20000	0.28195
0.05729	17.45454	0.29215
0.06250	16.00000	0.30325
0.06771	14.76923	0.31607
0.07292	13.71429	0.33195
0.07812	12.80001	0.35343
0.08333	12.00000	0.38606
0.08854	11.29412	0.44398
0.09375	10.66667	0.56188
0.09896	10.10526	0.73148
0.10417	9.60000	0.83074
0.10937	9.14286	0.87483
0.11458	8.72727	0.89753
0.11979	8.34783	0.91110
0.12500	8.00000	0.92016
0.13021	7.68000	0.92671
0.13542	7.38462	0.93177
0.14062	7.11111	0.93588
0.14583	6.85715	0.93936
0.15104	6.62069	0.94243
0.15625	6.40000	0.94522
0.16146	6.19355	0.94784
0.16667	6.00000	0.95038
0.17187	5.81819	0.95290
0.17708	5.64706	0.95545
0.18229	5.48572	0.95808
0.18750	5.33334	0.96083
0.19271	5.18919	0.96371
0.19792	5.05263	0.96669
0.20312	4.92308	0.96970
0.20833	4.80000	0.97261
0.21354	4.68293	0.97528
0.21875	4.57143	0.97763
0.22396	4.46512	0.97961
0.22917	4.36364	0.98124
0.23437	4.26667	0.98258



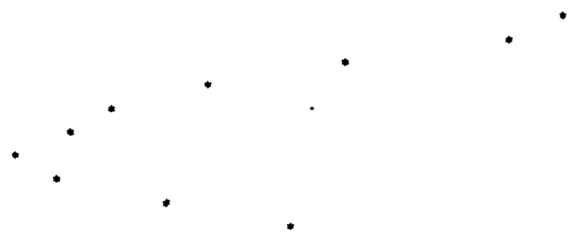
0.23958	4.17392	0.98367	*
0.24479	4.08511	0.98458	*
0.25000	4.00000	0.98535	*
0.25521	3.91837	0.98600	*
0.26042	3.84000	0.98657	*
0.26562	3.76471	0.98707	*
0.27083	3.69231	0.98751	*
0.27604	3.62264	0.98792	*
0.28125	3.55556	0.98830	*
0.28646	3.49091	0.98866	*
0.29167	3.42857	0.98900	*
0.29687	3.36842	0.98933	*
0.30208	3.31035	0.98965	*
0.30729	3.25424	0.98997	*
0.31250	3.20000	0.99029	*
0.31771	3.14754	0.99061	*
0.32292	3.09678	0.99094	*
0.32812	3.04762	0.99127	*
0.33333	3.00000	0.99160	*
0.33854	2.95385	0.99194	*
0.34375	2.90909	0.99227	*
0.34896	2.86567	0.99260	*
0.35417	2.82353	0.99293	*
0.35937	2.78261	0.99324	*
0.36458	2.74286	0.99355	*
0.36979	2.70423	0.99383	*
0.37500	2.66667	0.99410	*
0.38021	2.63014	0.99435	*
0.38542	2.59459	0.99458	*
0.39062	2.56000	0.99480	*
0.39583	2.52632	0.99501	*
0.40104	2.49351	0.99521	*
0.40625	2.46154	0.99539	*
0.41146	2.43038	0.99557	*
0.41667	2.40000	0.99575	*
0.42187	2.37037	0.99592	*
0.42708	2.34146	0.99610	*
0.43229	2.31325	0.99627	*
0.43750	2.28571	0.99645	*
0.44271	2.25882	0.99664	*
0.44792	2.23256	0.99683	*
0.45312	2.20690	0.99704	*
0.45833	2.18182	0.99727	*
0.46354	2.15730	0.99751	*
0.46875	2.13333	0.99778	*
0.47396	2.10989	0.99808	*
0.47917	2.08696	0.99841	*
0.48437	2.06452	0.99877	*
0.48958	2.04255	0.99916	*
0.49479	2.02105	0.99958	*
0.50000	2.00000	1.00000	*

MOVING AVERAGE MODEL VIA SUBR. ARMA. RVAR= 0.14773

FIRST 10 COEFFICIENTS OF INFINITE MA:

I	BETA

1	1.2539
2	1.0977
3	0.6273
4	0.2406
5	-0.0592
6	-0.1896
7	-0.3284
8	-0.1990
9	0.1022
10	0.4530



LAG	RYY(V)	REY(V)	RYE(V)	REE(V)	PVH(V)
0	1.0000000	0.1477280	0.1477280	0.1477280	0.1477280
1	0.8241955	0.1852343	0.0	0.0	0.3799910
2	0.4610662	0.1621633	0.0	0.0	0.5580000
3	0.0643274	0.0926757	0.0	0.0	0.6161392
4	-0.2290547	0.0355393	0.0	0.0	0.6246889
5	-0.3617923	-0.0087525	0.0	0.0	0.6252075
6	-0.3095543	-0.0280102	0.0	0.0	0.6305184
7	-0.1034002	-0.0485089	0.0	0.0	0.6464470
8	0.1934883	-0.0293971	0.0	0.0	0.6522968
9	0.4531194	0.0151026	0.0	0.0	0.6538408
10	0.5753842	0.0669199	0.0	0.0	0.6841550
11	0.5210851	0.0943505	0.0	0.0	0.7444144
12	0.3289428	0.0931392	0.0	0.0	0.8031365
13	0.0765781	0.0665897	0.0	0.0	0.8331524
14	-0.1471764	0.0322674	0.0	0.0	0.8402004
15	-0.2750675	-0.0044234	0.0	0.0	0.8403328
16	-0.2702639	-0.0317387	0.0	0.0	0.8471517
17	-0.1442332	-0.0418515	0.0	0.0	0.8590083
18	0.0492022	-0.0294647	0.0	0.0	0.8648850
19	0.2329595	-0.0016271	0.0	0.0	0.8649029
20	0.3405738	0.0297803	0.0	0.0	0.8709062
21	0.3388526	0.0513266	0.0	0.0	0.8887391
22	0.2370471	0.0571676	0.0	0.0	0.9108618
23	0.0771126	0.0466486	0.0	0.0	0.9255921
24	-0.0824012	0.0247069	0.0	0.0	0.9297242
25	-0.1873946	-0.0012775	0.0	0.0	0.9297352
26	-0.2058632	-0.0221673	0.0	0.0	0.9330615
27	-0.1385778	-0.0311059	0.0	0.0	0.9396112
28	-0.0165965	-0.0255993	0.0	0.0	0.9440472
29	0.1118597	-0.0091053	0.0	0.0	0.9446084
30	0.2004484	0.0113967	0.0	0.0	0.9454876
31	0.2206798	0.0281378	0.0	0.0	0.9508470
32	0.1702797	0.0356167	0.0	0.0	0.9594340
33	0.0716521	0.0318422	0.0	0.0	0.9662974
34	-0.0376295	0.0188403	0.0	0.0	0.9687002
35	-0.1190166	0.0016216	0.0	0.0	0.9687179
36	-0.1465319	-0.0135378	0.0	0.0	0.9699585
37	-0.1151065	-0.0215335	0.0	0.0	0.9730973
38	-0.0407540	-0.0201678	0.0	0.0	0.9758505
39	0.0467212	-0.0107776	0.0	0.0	0.9766368
40	0.1153275	0.0026240	0.0	0.0	0.9766833
41	0.1422104	0.0149634	0.0	0.0	0.9781989
42	0.1209039	0.0219672	0.0	0.0	0.9814655
43	0.0623043	0.0215286	0.0	0.0	0.9846028
44	-0.0104446	0.0143001	0.0	0.0	0.9859870
45	-0.0709979	0.0032675	0.0	0.0	0.9860592
46	-0.0992993	-0.0074515	0.0	0.0	0.9864351
47	-0.0881484	-0.0141634	0.0	0.0	0.9877930
48	-0.0447187	-0.0148676	0.0	0.0	0.9892892
49	0.0130867	-0.0098285	0.0	0.0	0.9899431
50	0.0637510	-0.0012917	0.0	0.0	0.9899544
51	0.0899499	0.0074532	0.0	0.0	0.9903303
52	0.0842866	0.0133001	0.0	0.0	0.9915277
53	0.0511090	0.0143929	0.0	0.0	0.9929300
54	0.0040742	0.0106851	0.0	0.0	0.9937028
55	-0.0393670	0.0038191	0.0	0.0	0.9938015
56	-0.0643867	-0.0035774	0.0	0.0	0.9938881
57	-0.0637714	-0.0088929	0.0	0.0	0.9944234
58	-0.0398486	-0.0104457	0.0	0.0	0.9951620

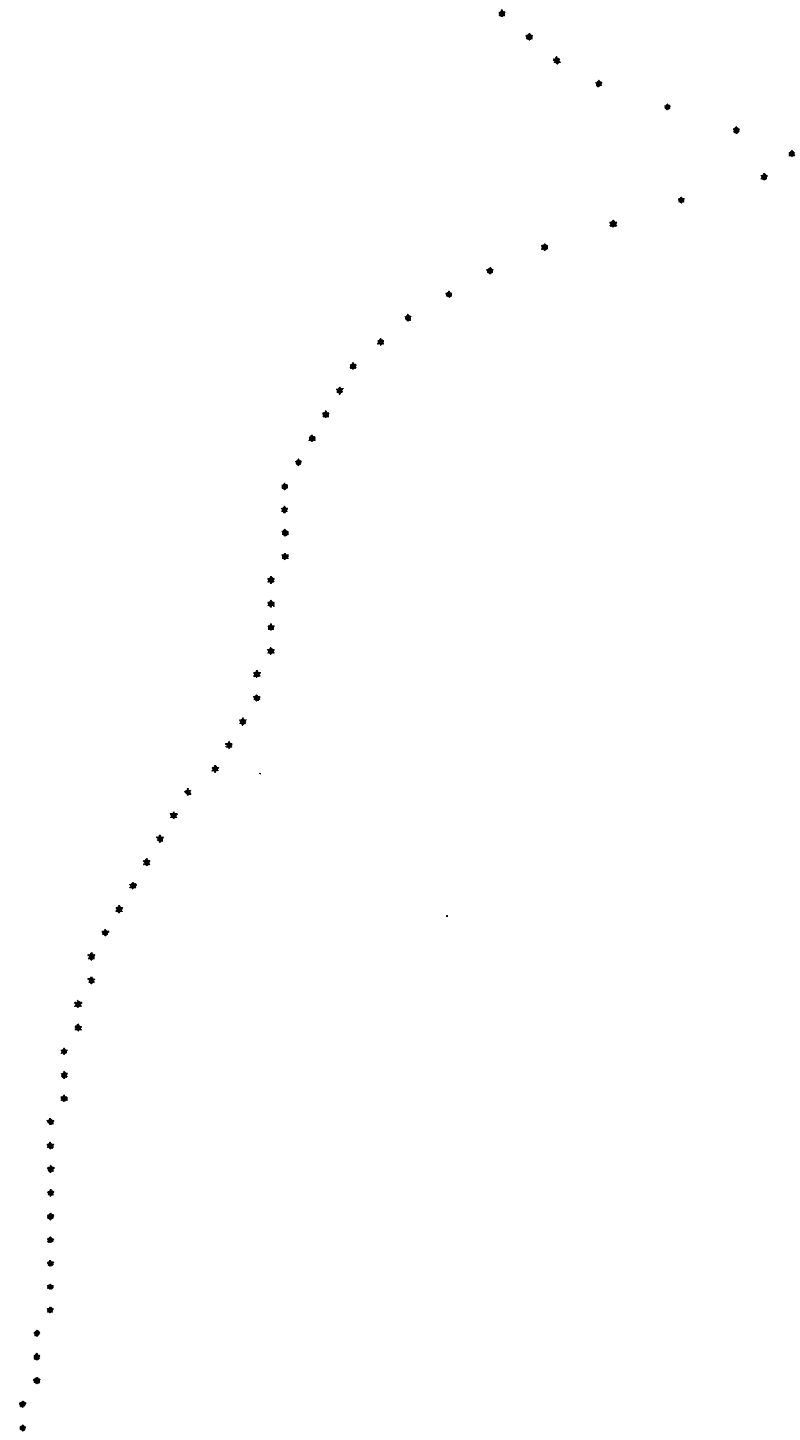
59	-0.0029164	-0.0080008	0.0	0.0	0.9955953
60	0.0329950	-0.0027385	0.0	0.0	0.9956460
61	0.0553327	0.0032664	0.0	0.0	0.9957182
62	0.0573010	0.0078396	0.0	0.0	0.9961342
63	0.0397851	0.0094755	0.0	0.0	0.9967419
64	0.0104273	0.0078145	0.0	0.0	0.9971552
65	-0.0195937	0.0036809	0.0	0.0	0.9972469
66	-0.0397664	-0.0012972	0.0	0.0	0.9972583
67	-0.0438760	-0.0053205	0.0	0.0	0.9974499
68	-0.0318346	-0.0070623	0.0	0.0	0.9977875
69	-0.0092135	-0.0060963	0.0	0.0	0.9980390
70	0.0151731	-0.0029766	0.0	0.0	0.9980990
71	0.0326239	0.0010313	0.0	0.0	0.9981061
72	0.0375922	0.0044488	0.0	0.0	0.9982401
73	0.0293675	0.0061181	0.0	0.0	0.9984934
74	0.0119194	0.0055833	0.0	0.0	0.9987044
75	-0.0078895	0.0031963	0.0	0.0	0.9987735
76	-0.0229555	-0.0000690	0.0	0.0	0.9987735
77	-0.0284468	-0.0030056	0.0	0.0	0.9988346
78	-0.0233025	-0.0046142	0.0	0.0	0.9989787
79	-0.0102774	-0.0044372	0.0	0.0	0.9991120
80	0.0053628	-0.0026775	0.0	0.0	0.9991605
81	0.0179151	-0.0000762	0.0	0.0	0.9991605
82	0.0232823	0.0023875	0.0	0.0	0.9991990
83	0.0202608	0.0038587	0.0	0.0	0.9992998
84	0.0106951	0.0038957	0.0	0.0	0.9994025
85	-0.0014568	0.0025937	0.0	0.0	0.9994479
86	-0.0117127	0.0005095	0.0	0.0	0.9994497
87	-0.0166970	-0.0015686	0.0	0.0	0.9994663
88	-0.0152256	-0.0029149	0.0	0.0	0.9995238
89	-0.0085039	-0.0031159	0.0	0.0	0.9995894
90	0.0005594	-0.0021908	0.0	0.0	0.9996219
91	0.0085653	-0.0005517	0.0	0.0	0.9996240
92	0.0128759	0.0011709	0.0	0.0	0.9996332
93	0.0124425	0.0023659	0.0	0.0	0.9996710
94	0.0079964	0.0026543	0.0	0.0	0.9997187
95	0.0015942	0.0020036	0.0	0.0	0.9997458
96	-0.0042568	0.0007131	0.0	0.0	0.9997492
97	-0.0075950	-0.0007169	0.0	0.0	0.9997527
98	-0.0076569	-0.0017756	0.0	0.0	0.9997740
99	-0.0049806	-0.0021214	0.0	0.0	0.9998045
100	-0.0000466	-0.0016888	0.0	0.0	0.9998237
101	0.0028746	-0.0006899	0.0	0.0	0.9998270
102	0.0052827	0.0004813	0.0	0.0	0.9998285
103	0.0056995	0.0014007	0.0	0.0	0.9998417
104	0.0044208	0.0017651	0.0	0.0	0.9998628
105	0.0023895	0.0014890	0.0	0.0	0.9998778
106	0.0007181	0.0007180	0.0	0.0	0.9998812

REGRESSION ESTIMATION STAGEWISE SUMMARY
NUMBER OF VARIABLES IN FULL MODEL = 60
MAXIMUM NUMBER OF COEFFICIENTS = 0

VAR IN MODEL	COEFF				
CRIT ADD	CRIT DEL		AIC	FPE ADD	FPE DEL
23.610504	************	0.0		146.728394	***********
VAR ADD	VAR DEL	VAR LAST	NO. PREDS	RES VAR	NO. CYC
1	0	0	0	1.00000	0

```
VAR IN MODEL          COEFF
           1        0.82420
   CRIT ADD       CRIT DEL             AIC         FPE ADD          FPE DEL
  16.240372     136.634933      -1.132591      100.483231      -455.404541
   VAR ADD        VAR DEL     VAR LAST     NO. PREDS       RES VAR       NO. CYC
         2              1            1             1       0.32070             1
--------------------------------------------------------------------------------
VAR IN MODEL          COEFF
           2       -0.68048
           1        1.38505
   CRIT ADD       CRIT DEL             AIC         FPE ADD          FPE DEL
   3.224512      55.499924      -1.749803       24.167328      -186.276840
   VAR ADD        VAR DEL     VAR LAST     NO. PREDS       RES VAR       NO. CYC
         8              2            2             2       0.17220             2
--------------------------------------------------------------------------------
VAR IN MODEL          COEFF
           8        0.15144
           2       -0.57455
           1        1.31340
   CRIT ADD       CRIT DEL             AIC         FPE ADD          FPE DEL
   0.699696       8.005162      -1.862652        3.126244       -27.055817
   VAR ADD        VAR DEL     VAR LAST     NO. PREDS       RES VAR       NO. CYC
         6              8            8             3       0.15311             3
--------------------------------------------------------------------------------


TO INTERPRET CRITICAL ADD AND DELETE VALUES, NOTE THAT
THEY ARE COMPUTED USING TWICE THE SAMPLE SIZE.


RESIDUAL VARIANCE FROM SUBSET ARMA =  0.1531089

    I       ALPH
---------------   .........................................
    1   -1.3134  *
    2    0.5746                                          *
    3    0.0                               *
    4    0.0                               *
    5    0.0                               *
    6    0.0                               *
    7    0.0                               *
    8   -0.1514                         *


SPECTRA, SPLMIN =     -4.01966  SPLMAX =      2.58945    SPLR =      6.60911

 FREQUENCY       PERIOD           SPEC
----------------------------------     ..........................................................
   0.0          0.0         12.71934                                                             *
   0.00521    192.00005     10.16380                                                           *
   0.01042     96.00002      6.41623                                                       *
   0.01562     64.00002      4.07023                                                   *
   0.02083     48.00000      2.78173                                                *
   0.02604     38.40001      2.05514                                             *
   0.03125     32.00000      1.62622                                           *
   0.03646     27.42857      1.36578                                         *
   0.04167     24.00000      1.20942                                        *
   0.04687     21.33333      1.12441                                        *
   0.05208     19.20000      1.09533                                       *
   0.05729     17.45454      1.11793                                        *
   0.06250     16.00000      1.19787                                        *
```

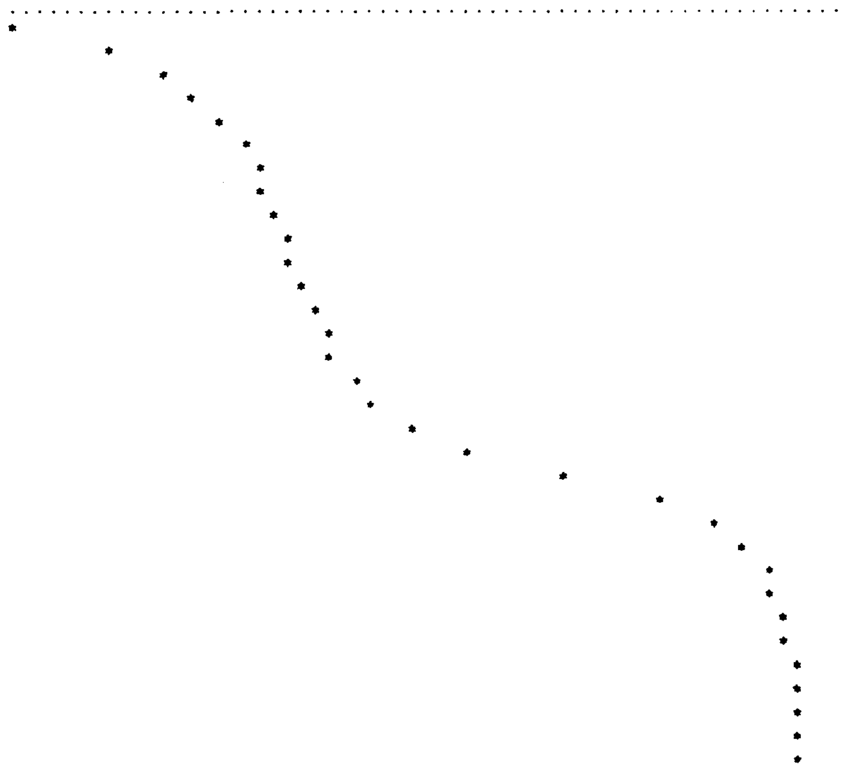
0.06771	14.76923	1.35349
0.07292	13.71429	1.62493
0.07812	12.80001	2.09766
0.08333	12.00000	2.96379
0.08854	11.29412	4.68247
0.09375	10.66667	8.23689
0.09896	10.10526	13.14637
0.10417	9.60000	11.07460
0.10937	9.14286	5.76709
0.11458	8.72727	3.02714
0.11979	8.34783	1.78178
0.12500	8.00000	1.16120
0.13021	7.68000	0.81942
0.13542	7.38462	0.61541
0.14062	7.11111	0.48599
0.14583	6.85715	0.40008
0.15104	6.62069	0.34118
0.15625	6.40000	0.29995
0.16146	6.19355	0.27082
0.16667	6.00000	0.25026
0.17187	5.81819	0.23595
0.17708	5.64706	0.22624
0.18229	5.48572	0.21983
0.18750	5.33334	0.21555
0.19271	5.18919	0.21225
0.19792	5.05263	0.20872
0.20312	4.92308	0.20377
0.20833	4.80000	0.19639
0.21354	4.68293	0.18603
0.21875	4.57143	0.17282
0.22396	4.46512	0.15750
0.22917	4.36364	0.14121
0.23437	4.26667	0.12511
0.23958	4.17392	0.11009
0.24479	4.08511	0.09667
0.25000	4.00000	0.03506
0.25521	3.91837	0.07523
0.26042	3.84000	0.06704
0.26562	3.76471	0.06030
0.27083	3.69231	0.05478
0.27604	3.62264	0.05029
0.28125	3.55556	0.04668
0.28646	3.49091	0.04379
0.29167	3.42857	0.04150
0.29687	3.36842	0.03972
0.30208	3.31035	0.03837
0.30729	3.25424	0.03736
0.31250	3.20000	0.03663
0.31771	3.14754	0.03611
0.32292	3.09678	0.03575
0.32812	3.04762	0.03546
0.33333	3.00000	0.03520
0.33854	2.95385	0.03488
0.34375	2.90909	0.03445
0.34896	2.86567	0.03386
0.35417	2.82353	0.03307
0.35937	2.78261	0.03209
0.36458	2.74286	0.03093
0.36979	2.70423	0.02964
0.37500	2.66667	0.02826
0.38021	2.63014	0.02685



0.38542	2.59459	0.02546
0.39062	2.56000	0.02414
0.39583	2.52632	0.02292
0.40104	2.49351	0.02182
0.40625	2.46154	0.02086
0.41146	2.43038	0.02005
0.41667	2.40000	0.01937
0.42187	2.37037	0.01883
0.42708	2.34146	0.01843
0.43229	2.31325	0.01816
0.43750	2.28571	0.01800
0.44271	2.25882	0.01796
0.44792	2.23256	0.01802
0.45312	2.20690	0.01817
0.45833	2.18182	0.01840
0.46354	2.15730	0.01869
0.46875	2.13333	0.01901
0.47396	2.10989	0.01936
0.47917	2.08696	0.01970
0.48437	2.06452	0.02000
0.48958	2.04255	0.02024
0.49479	2.02105	0.02039
0.50000	2.00000	0.02045

CUMULATIVE SUBSET ARMA SPECTRA :

FREQUENCY	PERIOD	SPEC
0.0	0.0	0.05865
0.00521	192.00005	0.16068
0.01042	96.00002	0.22786
0.01562	64.00002	0.27001
0.02083	48.00000	0.29820
0.02604	38.40001	0.31860
0.03125	32.00000	0.33446
0.03646	27.42857	0.34758
0.04167	24.00000	0.35905
0.04687	21.33333	0.36958
0.05208	19.20000	0.37972
0.05729	17.45454	0.38995
0.06250	16.00000	0.40077
0.06771	14.76923	0.41284
0.07292	13.71429	0.42712
0.07812	12.80001	0.44522
0.08333	12.00000	0.47024
0.08854	11.29412	0.50875
0.09375	10.66667	0.57510
0.09896	10.10526	0.68583
0.10417	9.60000	0.79833
0.10937	9.14286	0.86247
0.11458	8.72727	0.89544
0.11979	8.34783	0.91422
0.12500	8.00000	0.92613
0.13021	7.68000	0.93437
0.13542	7.38462	0.94046
0.14062	7.11111	0.94521
0.14583	6.85715	0.94907
0.15104	6.62069	0.95234
0.15625	6.40000	0.95520
0.16146	6.19355	0.95776



0.16667	6.00000	0.96011
0.17187	5.81819	0.96231
0.17708	5.64706	0.96442
0.18229	5.48572	0.96646
0.18750	5.33334	0.96846
0.19271	5.18919	0.97042
0.19792	5.05263	0.97235
0.20312	4.92308	0.97425
0.20833	4.80000	0.97608
0.21354	4.68293	0.97782
0.21875	4.57143	0.97944
0.22396	4.46512	0.98093
0.22917	4.36364	0.98227
0.23437	4.26667	0.98346
0.23958	4.17392	0.98451
0.24479	4.08511	0.98543
0.25000	4.00000	0.98624
0.25521	3.91837	0.98696
0.26042	3.84000	0.98759
0.26562	3.76471	0.98816
0.27083	3.69231	0.98868
0.27604	3.62264	0.98916
0.28125	3.55556	0.98959
0.28646	3.49091	0.99000
0.29167	3.42857	0.99039
0.29687	3.36842	0.99076
0.30208	3.31035	0.99112
0.30729	3.25424	0.99146
0.31250	3.20000	0.99180
0.31771	3.14754	0.99214
0.32292	3.09678	0.99247
0.32812	3.04762	0.99280
0.33333	3.00000	0.99312
0.33854	2.95385	0.99344
0.34375	2.90909	0.99376
0.34896	2.86567	0.99408
0.35417	2.82353	0.99438
0.35937	2.78261	0.99468
0.36458	2.74286	0.99497
0.36979	2.70423	0.99525
0.37500	2.66667	0.99551
0.38021	2.63014	0.99576
0.38542	2.59459	0.99600
0.39062	2.56000	0.99622
0.39583	2.52632	0.99644
0.40104	2.49351	0.99664
0.40625	2.46154	0.99684
0.41146	2.43038	0.99702
0.41667	2.40000	0.99720
0.42187	2.37037	0.99738
0.42708	2.34146	0.99755
0.43229	2.31325	0.99772
0.43750	2.28571	0.99788
0.44271	2.25882	0.99805
0.44792	2.23256	0.99821
0.45312	2.20690	0.99838
0.45833	2.18182	0.99855
0.46354	2.15730	0.99872
0.46875	2.13333	0.99890
0.47396	2.10989	0.99907
0.47917	2.08696	0.99925

0.48437	2.06452	0.99944
0.48958	2.04255	0.99962
0.49479	2.02105	0.99981
0.50000	2.00000	1.00000

ARSPIQ - AUTOREGRESSIVE SPECTRAL INFORMATION QUANTILE IDENTIFICATION

EMANUEL PARZEN, TERRY J. WOODFIELD, AND H. JOSEPH NEWTON
DEPARTMENT OF STATISTICS, TEXAS A&M UNIVERSITY,
COLLEGE STATION, TEXAS 77843
JULY 1983

MEMBER	NUMBER OF RECORDS
ARSPIO	4365

```
C     PROGRAM ARSPIO  (ARSP01)
C
C*****************************************************************
C
C     ARSPIO SPECTRAL ANALYSIS VERSION USING QUANTILES AND
C     INFORMATION FUNCTIONALS.     (CODE 1, JULY 1983)
C
C     THIS PROGRAM CARRIES OUT THE MEMORY ANALYSIS OF A TIME SERIES
C     AS DESCRIBED IN PARZEN (DEPARTMENT OF STATISTICS TECH. REPORT
C     NO. A-22, TEXAS A&M UNIVERSITY, JULY 1983)  A QUANTILE
C     ANALYSIS IS CARRIED OUT FOR THE FOLLOWING DATA BATCHES
C                         1  THE RAW DATA
C                         2  THE SAMPLE SPECTRAL DENSITY (PERIODOGRAM)
C                         3  THE CORRELOGRAM
C                         4  THE PARTIAL AUTOCORRELATIONS
C
C     THE CURRENT OPTIONS ARE:
C           NTAPE - TAPE NUMBER WHERE DATA RESIDES IN "ARSPIO FORMAT"
C           IRWND - 1 IF TAPE IS TO BE REWOUND, 0 OTHERWISE
C           NCOVM - NUMBER OF COVARIANCES TO BE COMPUTED
C           NFREQS - NUMBER OF FREQUENCIES FOR PERIODOGRAM (>=N+NCOVM)
C           IOPTMX - 2 IF SELECT ARMA MODELING DESIRED, 3 IF ARMA
C                       MODELING USING CEPSTRAL CORRELATIONS DESIRED,
C                       4 IF BOTH METHODS DESIRED, 1 OTHERWISE
C           NOMP - LENGTH OF NEIGHBORHOOD IN LOCAL QUANTILE ANALYSIS
C                   OF PERIODOGRAM  (=-1 IF NO LOCAL QUANT. ANALYSIS)
C           XSEAS - SEASONAL PERIOD, I.E., 12 FOR MONTHLY DATA
C           REXP - POWER TRANSFORMATION. IF REXP GT 1, LOG TRANS.
C
C     DEFAULT INPUT OPTIONS (PLACE A ZERO IN COLUMN FIVE OF OPTION
C     CARD): NTAPE=5    IRWND=0    NCOVM=N/2    NFREQS FOUND BY FNFREQ
C            XSEAS=12   IOPTMX=2   NOMP=N/XSEAS REXP=1.0
C     SPECIFYING A ZERO FOR ANY OPTION PRODUCES DEFAULT VALUE.
C
C     ARMA MODELING PERFORMED USING INFINITE MA REPRESENTATION OF
C     HIGH ORDER AR MODEL (ORDER=MAX(BEST ORDER, 2ND BEST ORDER)).
C     BURG'S ALGORITHM USED TO GENERATE AR COEFFICIENTS.
C     WITH IOPTMX=3 OR 4, CEPSTRAL CORRELATIONS USED TO FORM TRUNCATED
C     INFINITE ORDER MA TO PRODUCE COVARIANCE MATRIX FOR ARMA SELECT
C     ANALYSIS.
C
C     PROGRAMMER: TERRY J. WOODFIELD
C
C*****************************************************************
C
      COMMON /DAT/ YT(1000),CT(1200),ST(1200),FT(1200),WKM(61,61),
     &        IERR(500),R(250),CORR(250),INDT(60),COFT(60),
     &        AIC(250),ALRV(250),
     &        SIG(2),MINVC(2),NAMES(250),
     &        LABY(20),IC1(10),ALPHTP(250)
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION LL2(2,2)
      DIMENSION RCOEFF(6),X(650,6),WKM1(6,6),COVMAT(6,6)
      DIMENSION L(5),LABD(20)
C-----------------------------------------------------------------
      DATA LL2/4HBEST,4H    ,4H2ND ,4HBEST/
      DATA NIN/5/
      DATA LABD/'ORIG','INAL',' DAT','A   ',16*'    '/
      IUNIT=6
      NSCRCH=2
      TWOPI=8.*ATAN(1.0)
      WRITE(IUNIT,300)
      WRITE(IUNIT,5)
    5 FORMAT(//,T10,81(1H*),/,T10,'*',T90,'*',/,T10,
     +'*    ARSPIO - AUTOREGRESSIVE SPECTRAL INFORMATION QUANTILE ',
     +'IDENTIFICATION',T90,'*',/,T10,'*',T90,'*',/,T10,'*    EMANUEL ',
     +'PARZEN, TERRY J. WOODFIELD, AND H. JOSEPH NEWTON',T90,'*',/,
     +T10,'*    DEPARTMENT OF STATISTICS, TEXAS A&M UNIVERSITY,',
     +T90,'*',/,T10,'*    COLLEGE STATION, TEXAS 77843',T90,'*',/,
     +T10,'*    JULY 1983',T90,'*',/,T10,'*',T90,'*',/,T10,81(1H*))
C
C     INPUT NUMBER OF SERIES TO BE ANALYZED :
C
      READ(NIN,10) NSETS
   10 FORMAT(16I5)
C
C
C     START LOOP :
C
C
      DO 1300 NSET=1,NSETS
C
C     READ OVERALL OPTIONS :
C
      READ(NIN,20) NTAPE,IRWND,NCOVM,NFREQS,IOPTMX,NOMP,XSEAS,REXP
   20 FORMAT(6I5,2F5.0)
C
C     READ DATA SET :
C
      IF(NTAPE.EQ.0) NTAPE=5
      IF(IRWND.EQ.1) REWIND NTAPE
      CALL DATAIN(NTAPE,YT,NYT,L,LABY)
C
C     SET DEFAULT VALUES
C
      IF(NCOVM.EQ.0) NCOVM=(NYT/4)*2
      IF(NCOVM.GT.250) NCOVM=250
      IF(NFREQS.EQ.0) CALL FNFREQ(NYT,NCOVM,NFREQS)
      IF(XSEAS.EQ.0.0) XSEAS=12.0
      IF(IOPTMX.EQ.0) IOPTMX=2
      IF(NOMP.EQ.0) NOMP=IFIX(FLOAT(NYT)/XSEAS)
      IF(REXP.EQ.0.0) REXP=1.0
      IF(REXP.GT.1.0) GO TO 100
      IF(REXP.EQ.1.0) GO TO 120
      DO 90 I=1,NYT
      YT(I)=YT(I)**REXP
   90 CONTINUE
      GO TO 120
  100 CONTINUE
      DO 110 I=1,NYT
      YT(I)=ALOG(YT(I))
  110 CONTINUE
  120 CONTINUE
      WRITE(IUNIT,180) LABY,NTAPE,IRWND,NYT,NCOVM,NFREQS,XSEAS,NOMP,
     +REXP
  180 FORMAT(///,T10,20A4,/,T20,'INPUT TAPE = ',I3,T40,'IRWND = ',
     +I3,T60,'LENGTH OF SERIES = ',I6,/,T20,'NCOVM = ',I3,T40,
     +'NFREQS = ',I4,T60,'XSEAS = ',F6.2,/,T20,'NOMP = ',I3,T40,
     +'REXP = ',F6.3)
C
C     PERFORM ONESAM ANALYSIS OF TIME SERIES
C
      CALL QUENT(NYT,YT,LABY,LABD,1,0,NYT+4,Y25,Y50,Y75)
```

```
C
C     DETREND BY SUBTRACTING MEDIAN AND "NORMALIZE" BY DIVIDING
C     BY TWICE THE IQ RANGE.
C
      YMULT=1./(2.*(Y75-Y25))
      DO 190 I=1,NYT
  190 YT(I)=(YT(I)-Y50)*YMULT
C
C     DESCRIBE Y-TILDA :
C
  244 CONTINUE
      IF(NCOVM.NE.0)CALL DESYT(NYT,NFREQS,XSEAS,NCOVM,NOMP,
     &LL2,IOPTME)
      IF(NSET.NE.NSETS) WRITE(IUNIT,300)
  300 FORMAT(1H1)
C
C     FINISH LOOP :
C
 1300 CONTINUE
      WRITE(IUNIT,5)
      STOP
      END
      SUBROUTINE AICLS(RV,M,N,ALRV,AIC,LAIC)
C***********************************************************************
C
C     SUBROUTINE TO CALCULATE AKAIKES AIC CRITERION AIC(K)=
C     ALOG(RV(K))+2K/N AND ALSO ALRV(K)=ALOG(RV(K)),K=1,....,M.
C     FURTHER, LAIC=K THAT MINIMIZES AIC IS DETERMINED.
C
C***********************************************************************
C
      DIMENSION RV(M),AIC(M),ALRV(M)
C
C
      ON=N
      DO 10 K=1,M
      ALRV(K)=ALOG(RV(K))
      AIC(K)=ALRV(K)+((2.*FLOAT(K))/ON)
   10 CONTINUE
      CALL MIN(AIC,M,CC,LAIC)
      IF(CC.GT.0.0) LAIC=0
C
C
      RETURN
      END
      SUBROUTINE ARMA(ALPH,NP,NO,BETA)
C*****************************************************************
C
C     PURPOSE              - TO FIND THE COEFFICIENTS BETA(1),....,
C                              BETA(NO) OF A MOVING AVERAGE REPRESENTATION
C                              OF ORDER NO GIVEN THE AUTOREGRESSIVE
C                              COEFFICIENTS ALPH(1),...,ALPH(NP)
C
C     ARGUMENTS     ALPH   - INPUT VECTOR OF LENGTH NP CONTAINING THE
C                              COEFFICIENTS OF THE AUTOREGRESSIVE PROCESS
C                              OF ORDER NP
C                   NP     - ORDER OF THE AUTOREGRESSIVE PROCESS
C                   BETA   - OUTPUT VECTOR OF LENGTH NO CONTAINING THE
C                              COEFFICIENTS OF THE MOVING AVERAGE PROCESS
C                              OF ORDER NO
C                   NO     - ORDER OF THE MOVING AVERAGE PROCESS
C
C      REQD. TIMESBOARD SUBROUTINES - NONE
C
C*****************************************************************
C
      DIMENSION ALPH(NP),BETA(NO)
C
      BETA(1)=-ALPH(1)
      IF(NO.EQ.1) RETURN
      DO 1 I=2,NO
         C=0.
      IF(I.GT.NP) GO TO 2
         C=-ALPH(I)
    2    M=MIN0(NP,I-1)
         DO 3 J=1,M
    3    C=C-ALPH(J)*BETA(I-J)
    1 BETA(I)=C
C
      RETURN
      END
      SUBROUTINE ARSP(ALPH,SIG,NP,NFREQS,CT,SPEC)
C****************************************************************
C
C     SUBROUTINE TO CALCULATE THE AUTOREGRESSIVE SPECTRAL
C     DENSITY (AT THE NFREQS EQUALLY SPACED FREQUENCIES
C     BETWEEN 0 AND TWOPI) FOR THE AUTOREGRESSIVE PARAMETERS
C     ALPH(1),....ALPH(NP),AND SIG (RESIDUAL VARIANCE).
C
C     INPUT :
C             NP,ALPH(1),....,ALPH(NP),SIG,NFREQS
C
C     OUTPUT :
C             SPEC(1),....SPEC(NFREQS)
C
C     AUXILLIARY :
C             CT
C
C     SUBROUTINES CALLED : FFT
C
C****************************************************************
C
      DIMENSION ALPH(1),CT(NFREQS),SPEC(NFREQS)
C
      FAC=SIG/(8.*ATAN(1.0))
      IF(NP.GT.0) GO TO 10
      DO 5 I=1,NFREQS
    5 SPEC(I)=FAC
      GO TO 99
   10 CONTINUE
C
      DO 1 I=1,NFREQS
      CT(I)=0.
    1 SPEC(I)=0.
      CT(1)=1.
      DO 2 I=1,NP
    2 CT(I+1)=ALPH(I)
C
      CALL FFT(CT,SPEC,NFREQS,NFREQS,NFREQS,1)
C
      DO 3 I=1,NFREQS
    3 SPEC(I)=FAC/(CT(I)*CT(I)+SPEC(I)*SPEC(I))
C
   99 CONTINUE
      RETURN
      END
      SUBROUTINE CEPSTC(FT,NFREQS,NCOVM,CT,ST,SIG)
C***********************************************************************
C
C     SUBROUTINE TO CALCULATE THE CEPSTRAL CORRELATIONS GIVEN
C     THE SPECTRAL DENSITY FUNCTION FT(1),....,FT(NFREQS).
C
```

```
C     INPUT   NFREQS - NUMBER OF EQUALLY SPACED FREQUENCIES
C                      BETWEEN 0 AND TWOPI
C             FT - SPECTRAL DENSITY FUNCTION
C             NCOVM - NUMBER OF CEPSTRAL CORRELATIONS DESIRED
C
C     OUTPUT: CT - CEPSTRAL CORRELATIONS
C              SIG - SIGMA INFINITY SQUARED FROM INTEGRAL LOG SPECTRUM
C
C     AUXILLIARY  ST
C
C     SUBPROGRAMS CALLED: FORIER
C
C***************************************************************
      DIMENSION FT(NFREQS),CT(NFREQS),ST(NFREQS)
      COMPLEX A(1200)
      FAC=1./FLOAT(NFREQS)
      DO 10 I=1,NFREQS
      CT(I)=ALOG(FT(I))
   10 ST(I)=FLOAT(I)*FAC
      NCVP1=NCOVM+1
      CALL FORIER(CT,ST,NFREQS,A,NCVP1)
      DO 20 I=1,NCOVM
   20 CT(I)=REAL(A(I+1))
      SIG=EXP(REAL(A(1)))
      RETURN
      END
      SUBROUTINE CHIP(IOPT,CHISQ)
      DIMENSION CHISQ(62),CHI(62)
      IF(IOPT.EQ.1) GO TO 10
      IF(IOPT.EQ.2) GO TO 20
      IF(IOPT.EQ.3) GO TO 30
   10 CONTINUE
      DATA CHI/6.64,9.22,11.32,13.28,
     115.09,16.81,18.47,20.08,
     121.65,23.19,24.75,26.25,27.72,29.17,30.61,32.03,33.44,
     134.83,36.22,37.59,38.96,40.31,41.66,43.00,44.34,45.66,
     146.99,48.30,49.61,50.91,52.21,53.51,54.80,56.08,57.36,
     158.64,59.91,61.18,62.45,63.71,64.97,66.23,67.48,68.73,
     169.98,71.22,72.46,73.70,74.94,76.17,77.40,78.63,79.86,
     181.09,82.31,83.53,84.75,85.97,87.18,88.40,89.61,90.82/
      GO TO 99
   20 CONTINUE
      DATA CHI/2.71,
     14.60,6.25,7.78,9.24,10.65,12.02,13.36,14.69,15.99,17.28,
     118.55,19.81,21.07,22.31,23.55,24.77,25.99,27.21,28.42,
     129.62,30.82,32.01,33.20,34.38,35.57,36.74,37.92,39.09,
     140.26,41.43,42.59,43.75,44.91,46.06,47.22,48.37,49.52,
     150.67,51.80,52.94,54.08,55.22,56.36,57.50,58.63,59.77,
     160.90,62.03,63.16,64.29,65.41,66.54,67.67,68.79,69.91,
     171.03,72.15,73.27,74.39,75.51,76.62/
      GO TO 99
   30 CONTINUE
      DATA CHI/1.64,
     13.22,4.64,5.99,7.29,8.56,9.80,11.30,12.24,13.44,14.63,
     115.81,16.99,18.15,19.31,20.47,21.62,22.76,23.90,25.04,
     126.17,27.30,28.43,29.55,30.68,31.80,32.91,34.03,35.14,
     136.25,37.36,38.47,39.57,40.68,41.78,42.88,43.98,45.08,
     146.17,47.26,48.36,49.45,50.54,51.63,52.72,53.81,54.90,
     155.99,57.07,58.16,59.24,60.32,61.41,62.49,63.57,64.65,
     165.73,66.81,67.89,68.97,70.04,71.12/
   99 CONTINUE
      DO 40 I=1,62
   40 CHISQ(I)=CHI(I)
      RETURN
      END
      SUBROUTINE CLPLTO(X,N,INIT,NAME,XMAX,XMIN)
C*************************************************************
C
C     SUBROUTINE TO PRINT AND PRINTER PLOT THE N-VECTOR X.
C     THIS ROUTINE IS A MODIFIED VERSION OF CLPLT1 DESIGNED TO
C     PLOT THE DELTA MEMORY FUNCTION.
C
C     INPUT :
C             N,X
C             INIT   PRINTED INDEX OF FIRST PRINTED X
C             NAME - 4 CHARACTER LITERAL CONSTANT GIVING
C                    LABEL FOR X
C             XMAX,XMIN : MAX AND MIN VALUES FOR SCALING PLOT
C
C     SUBROUTINES CALLED - NONE
C
C*************************************************************
C
      DIMENSION X(N),AL(101)
      DATA NOUT/6/
      DATA BLANK,DOT,Z,SL/1H ,1H.,1H*,1H|/
C
      IOPTX=0
      IF(N.GT.1) GO TO 10
      WRITE(NOUT,11) NAME,X(1)
   11 FORMAT(10X,A4,'(1) = ',F15.8)
      GO TO 99
   10 CONTINUE
C
C     INITIALIZE AL
C
      MM=61
      DN=(MM-1)/2
      DO 20 J=1,MM
   20 AL(J)=DOT
      WRITE(NOUT,25) NAME,(AL(J),J=1,MM)
   25 FORMAT(/,14X,1HI,6X,A4/10X,15(1H-),2X    1A1)
      DO 30 J=1,MM
   30 AL(J)=BLANK
C
C     FIND SCALING VALUES
C
      RX=XMAX-XMIN
      KZERO=31
      IF(RX.LT.1.E-20) IOPTX=1
C
C     PLOT
C
      JJ=INIT
      DO 40 J=1,N
      IF(IOPTX.EQ.1) GO TO 36
      C1=(X(J)-XMIN)/RX
      C1=2.*(C1-.5)
      GO TO 37
   36 C1=0.
   37 K=DN*(C1+1.)+1.5
      AL(KZERO)=SL
      AL(K)=Z
      WRITE(NOUT,38) JJ,X(J),(AL(I),I=1,MM)
   38 FORMAT(10X,I5,F10.4,2X,101A1)
      JJ=JJ+1
      AL(K)=BLANK
   40 CONTINUE
C
   99 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CLPLT1(X,N,INIT,NAME,MM,IOPT)                           ARSP10
C*****************************************************************      ARSP10
C                                                                        ARSP10
C   SUBROUTINE TO PRINT AND PRINTER PLOT THE N-VECTOR X.                 ARSP10
C                                                                        ARSP10
C   INPUT                                                                ARSP10
C           N,X                                                          ARSP10
C           INIT : PRINTED INDEX OF FIRST PRINTED X                      ARSP10
C           NAME : 4 CHARACTER LITERAL CONSTANT GIVING                   ARSP10
C                  LABEL FOR X                                           ARSP10
C           MM : NUMBER OF COLUMNS IN PLOT (LE.101)                      ARSP10
C           IOPT : 1,2 (POINT OR BAR PLOT)                               ARSP10
C                                                                        ARSP10
C   SUBROUTINES CALLED : MAX,MIN                                         ARSP10
C                                                                        ARSP10
C*****************************************************************      ARSP10
C                                                                        ARSP10
      DIMENSION X(N),AL(101)                                             ARSP10
      DATA NOUT/6/                                                       ARSP10
      DATA BLANK,DOT,Z/1H ,1H.,1H*/                                      ARSP10
C                                                                        ARSP10
      IOPTX=0                                                            ARSP10
      IF(N.GT.1) GO TO 10                                                ARSP10
      WRITE(NOUT,11) NAME,X(1)                                           ARSP10
   11 FORMAT(10X,A4,'(1) = ',F15.8)                                      ARSP10
      GO TO 99                                                           ARSP10
   10 CONTINUE                                                           ARSP10
C                                                                        ARSP10
C   INITIALIZE AL :                                                      ARSP10
C                                                                        ARSP10
      ON=(MM-1)/2                                                        ARSP10
      DO 20 J=1,MM                                                       ARSP10
   20 AL(J)=DOT                                                          ARSP10
      WRITE(NOUT,25) NAME,(AL(J),J=1,MM)                                 ARSP10
   25 FORMAT(/,14X,1HI,6X,A4/10X,15(1H-),2X,101A1)                       ARSP10
      DO 30 J=1,MM                                                       ARSP10
   30 AL(J)=BLANK                                                        ARSP10
C                                                                        ARSP10
C   FIND MAX AND MIN                                                     ARSP10
C                                                                        ARSP10
      CALL MAX(X,N,XMAX,IND)                                             ARSP10
      CALL MIN(X,N,XMIN,IND)                                             ARSP10
      RX=XMAX-XMIN                                                       ARSP10
      IF(RX.LT.1.E-20) IOPTX=1                                           ARSP10
C                                                                        ARSP10
C   PLOT                                                                 ARSP10
C                                                                        ARSP10
      JJ=INIT                                                            ARSP10
      DO 40 J=1,N                                                        ARSP10
      IF(IOPTX.EQ.1) GO TO 36                                            ARSP10
      C1=(X(J)-XMIN)/RX                                                  ARSP10
      C1=2.*(C1-.5)                                                      ARSP10
      GO TO 37                                                           ARSP10
   36 C1=0.                                                              ARSP10
   37 K=ON*(C1+1.)+1.5                                                   ARSP10
      AL(K)=Z                                                            ARSP10
      IF(IOPT.EQ.1) GO TO 35                                             ARSP10
      DO 39 I=1,K                                                        ARSP10
   39 AL(I)=Z                                                            ARSP10
   35 CONTINUE                                                           ARSP10
      WRITE(NOUT,38) JJ,X(J),(AL(I),I=1,MM)                              ARSP10
   38 FORMAT(10X,I5,F10.4,2X,101A1)                                      ARSP10
      JJ=JJ+1                                                            ARSP10
      AL(K)=BLANK                                                        ARSP10
      IF(IOPT.EQ.1) GO TO 40                                             ARSP10
      DO 41 I=1,K                                                        ARSP10
   41 AL(I)=BLANK                                                        ARSP10
   40 CONTINUE                                                           ARSP10
C                                                                        ARSP10
   99 CONTINUE                                                           ARSP10
      RETURN                                                             ARSP10
      END                                                                ARSP10
      SUBROUTINE CLPLT2(X,Y,N,INIT,NAMEX,NAMEY,MM,IOPT)                  ARSP10
C*****************************************************************      ARSP10
C                                                                        ARSP10
C   SUBROUTINE TO PRINT AND PRINTER PLOT THE N-VECTORS X,Y.              ARSP10
C                                                                        ARSP10
C   INPUT :                                                              ARSP10
C           N,X,Y                                                        ARSP10
C           INIT : PRINTED INDEX OF FIRST PRINTED X AND Y                ARSP10
C           NAMEX,NAMEY : 4CHARACTER LITERAL CONSTANTS                   ARSP10
C                         GIVING LABEL FOR X AND Y                       ARSP10
C           MM : NUMBER OF COLUMNS IN PLOT (LE.91)                       ARSP10
C           IOPT : 1,2 (POINT OR BAR PLOTS)                              ARSP10
C                                                                        ARSP10
C   SUBROUTINES CALLED : MAX,MIN                                         ARSP10
C                                                                        ARSP10
C*****************************************************************      ARSP10
C                                                                        ARSP10
      DIMENSION X(N),Y(N),AL(91)                                         ARSP10
      DATA NOUT,BLANK,DOT,Z1,Z2/6,1H ,1H.,1H*,1HX/                       ARSP10
C                                                                        ARSP10
      IF(N.GT.1) GO TO 10                                                ARSP10
      WRITE(NOUT,11) NAMEX,X(1),NAMEY,Y(1)                               ARSP10
   11 FORMAT(14X,A4,'(1) = ',F15.8,6X,A4,'(1) = ',F15.8)                 ARSP10
      GO TO 99                                                           ARSP10
   10 CONTINUE                                                           ARSP10
C                                                                        ARSP10
C   INITIALIZE AL :                                                      ARSP10
C                                                                        ARSP10
      IOPTX=0                                                            ARSP10
      IOPTY=0                                                            ARSP10
      I1=(MM/2)+1                                                        ARSP10
      ON=(MM-1)/2                                                        ARSP10
      DO 20 J=1,MM                                                       ARSP10
   20 AL(J)=DOT                                                          ARSP10
      WRITE(NOUT,25) NAMEX,NAMEY,(AL(J),J=1,MM)                          ARSP10
   25 FORMAT(/,14X,1HI,6X,A4,6X,A4/10X,25(1H-),2X,91A1)                  ARSP10
      DO 30 J=1,MM                                                       ARSP10
   30 AL(J)=BLANK                                                        ARSP10
C                                                                        ARSP10
C   FIND MAX AND MIN .                                                   ARSP10
C                                                                        ARSP10
      CALL MAX(X,N,XMAX,IND)                                             ARSP10
      CALL MIN(X,N,XMIN,IND)                                             ARSP10
      CALL MAX(Y,N,YMAX,IND)                                             ARSP10
      CALL MIN(Y,N,YMIN,IPD)                                             ARSP10
      RX=XMAX-XMIN                                                       ARSP10
      RY=YMAX-YMIN                                                       ARSP10
      IF(RX.LT.1.E-20) IOPTX=1                                           ARSP10
      IF(RY.LT.1.E-20) IOPTY=1                                           ARSP10
C                                                                        ARSP10
C   PLOT .                                                               ARSP10
C                                                                        ARSP10
      JJ=INIT                                                            ARSP10
      DO 45 J=1,N                                                        ARSP10
      IF(IOPTX.EQ.1) GO TO 36                                            ARSP10
      C1=((X(J)-XMIN)/RX)-1                                              ARSP10
      GO TO 37                                                           ARSP10
   36 C1=-.5                                                             ARSP10
   37 IF(IOPTY.EQ.1) GO TO 38                                            ARSP10
      C2=(Y(J)-YMIN)/RY                                                  ARSP10
```

```
      GO TO 39
   38 C2=.5
   39 K1=DN*(C1+1.)+1.5
      K2=DN*(C2+1.)+1.5
      AL(K1)=Z1
      AL(K2)=Z2
      IF(IOPT.EQ.1) GO TO 41
      DO 42 I=1,K1
   42 AL(I)=Z1
      DO 43 I=I1,K2
   43 AL(I)=Z2
   41 CONTINUE
      WRITE(NOUT,40) JJ,X(J),Y(J),(AL(I),I=1,MM)
   40 FORMAT(10X,I5,2F10.4,2X,91A1)
      JJ=JJ+1
      AL(K1)=BLANK
      AL(K2)=BLANK
      IF(IOPT.EQ.1) GO TO 45
      DO 44 I=1,K1
   44 AL(I)=BLANK
      DO 46 I=I1,K2
   46 AL(I)=BLANK
   45 CONTINUE
C
C
   99 CONTINUE
      RETURN
      END
      SUBROUTINE CPMA(FT,NFREQS,NCOVM,CT,ST,BETA,RVMA,SIG)
C***************************************************************
C
C   SUBROUTINE TO COMPUTE CEPSTRAL CORRELATIONS FROM SPECTRAL
C   DENSITY FUNCTION FT VIA CEPSTC AND THEN TO COMPUTE NCOVM
C   BETAS FOR AN INFINITE MOVING AVERAGE MODEL TRUNCATED TO
C   ORDER NCOVM.
C
C   INPUT: FT - SPECTRAL DENSITY FUNCTION EVALUATED AT NFREQS
C               EQUALLY SPACED FREQUENCIES BETWEEN 0 AND TWO PI.
C          NFREQS - NUMBER OF FREQUENCIES
C          NCOVM - NUMBER OF CEPSTRAL CORRELATIONS TO COMPUTE AND
C                  ORDER OF TRUNCATED MOVING AVERAGE MODEL
C
C   OUTPUT: CT - CEPSTRAL CORRELATIONS
C           BETA - VECTOR OF MOVING AVERAGE COEFFICIENTS
C           RVMA - RESIDUAL VARIANCE FOR MA MODEL
C           SIG - SIGMA INFINITY SQUARED FROM INTEGRAL LOG SPECTRUM
C
C   AUXILLIARY: ST
C
C   SUBPROGRAMS CALLED: CEPSTC,FFT,CLPLT1
C
C*****************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION FT(1),CT(1),ST(1),BETA(1)
      CALL CEPSTC(FT,NFREQS,NCOVM,CT,ST,SIG)
C
C   CT CONTAINS CEPSTRAL CORRELATIONS
C
      WRITE(IUNIT,10) SIG
   10 FORMAT(/,10X,'SIGMA INFINITY SQUARED (VIA SMOOTHED PER.) = ',
     *F10.5)
      CALL CLPLT1(CT,NCOVM,1,4HCEPC,41,1)
C
C   COMPUTE THE BETAS
C
      BETA(1)=CT(1)
      DO 825 I=2,NCOVM
      BETA(I)=0.
      IM1=I-1
      DO 820 K=1,IM1
  820 BETA(I)=BETA(I)+CT(K)*BETA(I-K)*FLOAT(K)
  825 BETA(I)=BETA(I)/FLOAT(I) + CT(I)
      RVMA=1.
      DO 830 I=1,NCOVM
  830 RVMA=RVMA+BETA(I)*BETA(I)
      RVMA=1./RVMA
      WRITE(IUNIT,832) RVMA
  832 FORMAT(//,10X,'MA MODEL VIA CEPSTRAL CORR., RVAR=',
     *F10.5,//,10X,'FIRST 10 COEFFICIENTS OF INFINITE MA:',//)
      CALL CLPLT1(BETA,10,1,4HBETA,41,1)
      RETURN
      END
      SUBROUTINE CUMSP(SPEC,NFREQS,IOPT,WK,N1)
C*********************************************************
C
C   SUBROUTINE TO PRINTER PLOT CUM SPEC DIST FCTN
C
C   INPUT :
C              NFREQS,SPEC(1),...,SPEC(NFREQS)
C              IF NFREQS.LT.0, CUM SPECTRA IS CALCULATED
C              BUT NOT PLOTTED.
C              IOPT : 1 MEANS FREQS ARE 0 TO 2PI
C                     0 MEANS FREQS ARE 0 TO PI
C
C   OUTPUT :
C              WK(1),...,WK(N1) WHERE N1=(NFREQS/2)+1 IF
C                  IOPT=1 OR NFREQS IF IOPT=0 : CUM DIST
C
C*********************************************************
C
      DIMENSION SPEC(1),WK(1)
C
C
      NFRQ=NFREQS
      NFREQS=IABS(NFREQS)
      FFREQ=0.0
      N1=(NFREQS/2)+1
      DELF=8.*ATAN(1.0)/FLOAT(NFREQS)
      IF(IOPT.EQ.1) GO TO 10
      N1=NFREQS
      DELF=8.*ATAN(1.0)/FLOAT(2*(NFREQS-1))
   10 CONTINUE
      C=SPEC(1)
      WK(1)=SPEC(1)
      DO 20 I=2,N1
      WK(I)=WK(I-1)+SPEC(I)
      C=C+SPEC(I)
   20 CONTINUE
      DO 30 I=1,N1
      WK(I)=WK(I)/C
   30 CONTINUE
C
C
      IF(NFRQ.LT.0) GO TO 40
      CALL SPPLT(WK,N1,FFREQ,DELF,0)
   40 NFREQS=NFRQ
C
C
      RETURN
      END
      SUBROUTINE CVARP(CORR,M,NP,WK1,ALPH)
C*********************************************************
```

```
C
C     SUBROUTINE TO CALCULATE THE COEFFICIENTS ALPH(1),...
C     ALPH(NP) OF AN AUTOREGRESSIVE PROCESS OF ORDER NP
C     GIVEN CORR(1),...,CORR(M). (M GE NP)
C
C     INPUT :
C              NP,M,CORR(1),...,CORR(M)
C
C     OUTPUT :
C              ALPH(1),...,ALPH(NP)
C
C     SUBROUTINES CALLED : NONE
C
C*****************************************************
C
      DIMENSION CORR(M),WK1(NP),ALPH(NP)
C
      WK1(1)=-CORR(1)
      IF(NP.GT.1) GO TO 10
      ALPH(1)=WK1(1)
      RETURN
C
   10 DO 20 I=2,NP
      C1=CORR(I)
      C2=1.
      IM1=I-1
         DO 25 J=1,IM1
         C1=C1+WK1(J)*CORR(I-J)
   25    C2=C2+WK1(J)*CORR(J)
      C=-(C1/C2)
      ALPH(I)=C
         DO 26 J=1,IM1
   26    ALPH(J)=WK1(J)+C*WK1(I-J)
         DO 27 J=1,I
   27    WK1(J)=ALPH(J)
   20 CONTINUE
C
      RETURN
      END
      SUBROUTINE CVCR(R,RO,M)
      DIMENSION R(M)
      DO 1 I=1,M
    1 R(I)=R(I)/RO
      RETURN
      END
      SUBROUTINE CVPART(R,RO,M,WK1,WK2,PART)
C*****************************************************
C
C     SUBROUTINE TO CALCULATE THE PARTIAL AUTOCORRELATIONS
C     PART(1),...,PART(M) GIVEN THE AUTOCOVARIANCES RO,
C     R(1),...,R(M).
C
C     INPUT :
C              M,RO,R(1),...,R(M)
C
C     OUTPUT :
C              PART(1),...,PART(M)
C
C     SUBROUTINES CALLED : NONE
C
C*****************************************************
C
      DIMENSION R(M),WK1(M),WK2(M),PART(M)
C
      DO 1 I=1,M
    1 R(I)=R(I)/RO
C
      WK1(1)=-R(1)
      PART(1)=-R(1)
      IF(M.EQ.1) GO TO 99
      DO 10 I=2,M
      C1=R(I)
      C2=1
         IM1=I-1
         DO 15 J=1,IM1
         C1=C1+WK1(J)*R(I-J)
   15    C2=C2+WK1(J)*R(J)
      PART(I)=-(C1/C2)
      WK2(I)=PART(I)
         DO 16 J=1,IM1
   16    WK2(J)=WK1(J)+PART(I)*WK1(I-J)
         DO 27 J=1,I
   27    WK1(J)=WK2(J)
   10 CONTINUE
C
   99 DO 100 I=1,M
  100 R(I)=R(I)*RO
C
      RETURN
      END
      SUBROUTINE CVSPW(R,RO,IOPT,NTRUNC,NFREQ,WORK1,WORK2,SPEC)
C.....................................................
C     GIVEN RO,R(1),...,R(NTRUNC), THE FIRST NTRUNC+1
C     AUTOCOVARIANCES OF A TIME SERIES X(.), SUBROUTINE CVSPW
C     CALCULATES SPEC(1),...,SPEC(NFREQ), A SMOOTHED ESTIMATE
C     OF THE SPECTRAL DENSITY OF X(.) AT THE NFREQ POINTS
C     0,TWOPI/NFREQ,...,TWOPI*(NFREQ-1)/NFREQ.  THE WEIGHTING
C     FUNCTION IS SPECIFIED BY THE USER BY IOPT (SEE BELOW)
C
C     METHOD : THE CHOSEN WEIGHTING FUNCTION WORK1(1),...
C      WORK1(NTRUNC) IS CALCULATED FOR A SPECIFIED TRUNCATION
C     POINT NTRUNC.  THEN A VECTOR OF LENGTH NFREQ IS
C     DETERMINED AS (RO,WORK1(1)*R(1),...,WORK1(NTRUNC)*
C     R(NTRUNC),0,...,0)
C     THEN SUBROUTINE FFT IS USED TO FIND THE COSINE
C     TRANSFORM OF THIS VECTOR. THIS TRANSFORM IS DIVIDED BY
C     TWOPI GIVING THE DESIRED SPECTRUM.
C
C     INPUT :
C     NTRUNC, NFREQ; RO,R(1),...,R(NTRUNC)
C
C     IOPT :
C
C     AS GIVEN BY T.W.ANDERSON : THE STAT ANAL OF TIME
C     SERIES, THE FOLLOWING OPTIONS OF WEIGHTING FUNCTIONS
C     ARE AVAILABLE .
C
C     IOPT=1 : BARTLETT
C     IOPT=2 : MODIFIED BARTLETT
C     IOPT=3 : HANNING
C     IOPT=4 : HAMMING
C     IOPT=5 : PARZEN(I)
C     IOPT=6 : PARZEN(J)  (NTRUNC MUST BE EVEV)
C
C     OUTPUT :
C     SPEC(1),...,SPEC(NFREQ)
C
C     AUXILLIARY : WORK1,WORK2
C
C     SUBROUTINES CALLED : FFT
C
C.....................................................
```

```
      DIMENSION R(NTRUNC),WORK1(NTRUNC),WORK2(NFREQ),SPEC(NFREQ)
C
C   INITIALIZE :
C
      PI=4.*ATAN(1.0)
      TWOPI=2.*PI
      DO 1 I=1,NFREQ
      SPEC(I)=0.
    1 WORK2(I)=0.
      SPEC(1)=R0
      C1=NTRUNC
      C2=NFREQ
C
C   CALCULATE SPECIFIED WEIGHTS :
C
      IF(IOPT.EQ.1) GO TO 2
      IF(IOPT.EQ.2) GO TO 4
      IF(IOPT.EQ.3) GO TO 6
      IF(IOPT.EQ.4) GO TO 8
      IF(IOPT.EQ.5) GO TO 10
      IF(IOPT.EQ.6) GO TO 12
      IF(IOPT.EQ.7) GO TO 20
      IF(IOPT.EQ.8) GO TO 22
      GO TO 99
C
C
C   BARTLETT :
C
    2 DO 3 J=1,NTRUNC
      D1=J
      D2=D1/C1
      D3=D1/C2
    3 WORK1(J)=2.*((1.-D2)/(1.-D3))
      GO TO 14
C
C   MODIFIED BARTLETT :
C
    4 DO 5 J=1,NTRUNC
    5 WORK1(J)=2.*(1.-(FLOAT(J)/C1))
      GO TO 14
C
C   HANNING :
C
    6 DO 7 J=1,NTRUNC
    7 WORK1(J)=1.+COS((PI*FLOAT(J))/C1)
      GO TO 14
C
C   HAMMING :
C
    8 DO 9 J=1,NTRUNC
    9 WORK1(J)=1.08+.92*COS((PI*FLOAT(J))/C1)
      GO TO 14
C
C   PARZEN(I) :
C
   10 DO 11 J=1,NTRUNC
   11 WORK1(J)=2.*(1.-(FLOAT(J)/C1)**2)
      GO TO 14
C
C   PARZEN(J) :
C
   12 M1=NTRUNC/2
      DO 13 J=1,M1
      J1=J+M1
      D1=FLOAT(J)/C1
      D2=D1**2
      D3=D1**3
      WORK1(J)=2.*(1.-6.*D2+6.*D3)
   13 WORK1(J1)=4.*(1.-(FLOAT(J1)/C1))**3
      GO TO 14
   20 DO 21 J=1,NTRUNC
   21 WORK1(J)=2.0/(1.0+(FLOAT(J)/C1)**4)
      GO TO 14
   22 DO 23 J=1,NTRUNC
   23 WORK1(J)=2.0/(1.0+(FLOAT(J)/C1)**8)
C
C   CALL FFT :
C
   14 DO 15 J=1,NTRUNC
   15 SPEC(J+1)=WORK1(J)*R(J)
      CALL FFT(SPEC,WORK2,NFREQ,NFREQ,NFREQ,1)
      DO 16 J=1,NFREQ
   16 SPEC(J)=SPEC(J)/TWOPI
   99 CONTINUE
      RETURN
      END
      SUBROUTINE DATAIN(NTAPE,X,N,L,LAB)
C*****************************************************************
C   SUBROUTINE TO READ A DATA FILE FROM TAPE NTAPE AS FOLLOWS :
C
C      CARD1 : LAB1(1),....,LAB(20)  (20A4)
C      CARD2 : SAMPLE SIZE N, FORMAT L(1),....,L(5),(I5,4X,5A4)
C      CARD3,CARD4,.... : DATA X(1),....,X(N) IN L FORMAT
C
C*****************************************************************
C
      DIMENSION X(1),L(5),LAB(20)
C
      READ(NTAPE,1) LAB
    1 FORMAT(20A4)
      READ(NTAPE,2) N,L
    2 FORMAT(I5,4X,5A4)
      READ(NTAPE,L) (X(I),I=1,N)
C
      RETURN
      END
      SUBROUTINE DESTAT(X,N,NAME,IHEAD,L,INIT,IOUT,Q25,Q50,Q75)
C*************************************************************************
C   SUBROUTINE TO PRINT ORDERED ARRAY BY QUANTILES AND COMPUTE
C   DESCRIPTIVE STATISTICS.
C   INPUT:
C        X: ARRAY OF ORDER STATISTICS
C        N: DIMENSION OF ARRAY X
C        NAME: NAME OF DATA SET.MUST BE ARRAY OF DIMENSION 20 IN
C              CALLING PROGRAM.
C        IUNIT: NUMBER OF UNIT OUTPUT IS DESIRED ON.
C        INIT: 0 FOR FIRST CALL, 1 THEREAFTER
C        IOUT: 1 IF QUANTILES TO BE LISTED, 0 OTHERWISE
C        WK1,WK2 : WORK VECTORS OF LENGTH NN1=N*2+3
C   OUTPUT: PRINTED OUTPUT IS ON IUNIT.
C   NO SUBROUTINES CALLED.
C*************************************************************************
      COMMON /UNIT/ IUNIT,NSCREN
      DIMENSION X(N),NAME(20),SUM(4),SUMSQ(4)
      DIMENSION L(4)
      DIMENSION IHEAD(20)
      NUNIT = IUNIT
C   COMPUTE L,THE ARRAY OF QUARTILE SIZES
      IF(INIT .EQ. 0) GOTO 5
      IF(L(4) .EQ. N) GOTO 25
5     LL = N/4
      L1 = LL
```

```
      L2 = LL
      L3 = LL
      L4 = LL
      IBRAN = MOD(N,4) + 1
      GOTO (20,11,12,13),IBRAN
   11 CONTINUE
      L4 = LL + 1
      GO TO 20
   12 CONTINUE
      L1 = LL + 1
      L4 = LL + 1
      GO TO 20
   13 CONTINUE
      L2 = LL + 1
      L3 = LL + 1
      L4 = LL + 1
   20 CONTINUE
      L(1)=L1
      L(2)=L(1) + L2
      L(3)=L(2) + L3
      L(4)=L(3) + L4
C     PRINT DATA ARRAY - ONE COLUMN FOR EACH QUARTER.
 25   WRITE(NUNIT,1001) NAME
      WRITE(NUNIT,1020) IHEAD
      IF(IOUT.EQ.0) GOTO 35
      WRITE(NUNIT,1002)
      WRITE(NUNIT,1003)
      DO 30 I = 1,LL
   30 WRITE(NUNIT,1004) I,X(I),X(L(1) + I),X(L(2)+I),X(L(3) + I)
      WRITE(NUNIT,1005)
      IF(L1 .GT. LL) WRITE(NUNIT,1006) X(L(1))
      IF(L2 .GT. LL) WRITE(NUNIT,1007) X(L(2))
      IF(L3 .GT. LL) WRITE(NUNIT,1008) X(L(3))
      IF(L4 .GT. LL) WRITE(NUNIT,1009) L4,X(L(4))
   35 IF(INIT .EQ. 1) RETURN
C     COMPUTE AND PRINT DESCRIPTIVE STATISTICS.
      K = 1
      S = 0
      SSO = 0.
      DO 50 I = 1,4
      SI = 0.0
      SOI = 0.0
      KK = L(I)
      DO 40 J = K,KK
      SI = SI + X(J)
      SOI = SOI + X(J)*X(J)
   40 CONTINUE
      K = 1 + L(I)
      S = S + SI
      SSO = SSO + SOI
      SUM(I) = SI
      SUMSO(I) = SOI
   50 CONTINUE
      IF(IOUT.EQ.0) GO TO 55
      WRITE(NUNIT,1010) (SUM   (I),I=1,4)
      WRITE(NUNIT,1011) (SUMSO (I),I=1,4)
   55 XBAR = S/FLOAT(N)
      VAR = (SSO -S*XBAR)/FLOAT(N-1)
      SD = SQRT(VAR)
C     IN THE ORIGINAL VERSION, O25,O50,O75 ARE COMPUTED HERE.
C     IN ARSPIO, SUB. QUENT COMPUTES THESE VALUES.
C
      OIO = O75 - O25
      XBARIO = (XBAR - O50) / (2. * OIO)
      SDIO   = SD / (2. * OIO)
      SDIOLG = ALOG(SDIO)
      SSON=SSO/FLOAT(N)
      WRITE(NUNIT,1012)
      WRITE(NUNIT,1013) N,O25,O50,O75,OIO
      WRITE(NUNIT,1014) SSON,XBAR,VAR,SD,XBARIO,SDIO,SDIOLG
  999 CONTINUE
      RETURN
 1001 FORMAT(///T20,20A4)
 1002 FORMAT(/T40,'ORDER STATISTICS IN QUARTERS'/T40,28(1H*))
 1003 FORMAT(T20,' SEQUENCE'/T20,'   WITHIN'/T20,' QUARTILE    ',
     +'FIRST QUARTER  SECOND QUARTER   THIRD QUARTER  FOURTH QUARTER'
     + /T21,8(1H*),2(3X,13(1H*),2X,14(1H*)))
 1004 FORMAT(T20,I8 , 4(1X,F15.4))
 1005 FORMAT(1X)
 1006 FORMAT(1H+,T29,F15.4)
 1007 FORMAT(1H+,T45,F15.4)
 1008 FORMAT(1H+,T61,F15.4)
 1009 FORMAT(1H+,T20,I8,T77,F15.4)
 1010 FORMAT(1H0,T20,'SUM',5X,4(1X,F15.4))
 1011 FORMAT(1H0,T20,'SUM OF'/T20,' SQUARES',1X,4(1X,F15.4))
 1012  FORMAT(///T45,'DESCRIPTIVE  STATISTICS'/T45,23(1H*))
 1013  FORMAT(//,T20,'SAMPLE',T28,'    LOWER',T52,'    UPPER   ',
     +          T64,'INT_QUARTL'/,T20,' SIZE'
     +          ,T28,' QUARTILE',T40,'  MEDIAN',T52,' QUARTILE',
     +          T64,'  RANGE',//,T20,I5,T29,4(G11.4,1X))
 1014  FORMAT(///,T15,'SUMSQ/N',T30,'MEAN',T40,' VARIANCE',T52,
     +          ' STD DEV',T66,'MEAN IO',T76,'STD DEV IO',T88,
     +          'LOG STD IO'//,T16,7(G11.4,1X))
 1020 FORMAT(T20,20A4)
      END
      SUBROUTINE DESYT(NYT,NFREQS,XSEAS,NCOVM,NQMP,
     &LL2,IOPTMX)
C*****************************************************************
C
C     SUBROUTINE TO DESCRIBE Y-TILDA VIA PERIODOGRAM, AUTOCORRELATIONS.
C     A ONESAM ANALYSIS OF PERIODOGRAM, CORRELOGRAM, AND PARTIAL
C     AUTOCORRELATIONS IS PERFORMED TESTING FOR WHITE NOISE.
C
C     INPUT: NYT - NUMBER OF OBSERVATIONS IN Y-TILDA
C            NFREQS - NUMBER OF EQUALLY SPACED FREQUENCIES AT
C                     WHICH TO OBTAIN SPECTRA
C            XSEAS - SEASONAL PERIOD OF DATA
C            NCOVM - NUMBER OF COVARIANCES TO COMPUTE
C            NQMP - SIZE OF NEIGHBORHOOD FOR LOCAL QUANTILE
C                   ANALYSIS OF PERIODOGRAM
C            LL2 - NAMES OF BEST, SECOND BEST AR MODELS
C            IOPTMX - SELECT MODELING OPTION
C                     1 - NO ARMA SELECT MODELING
C                     2 - ARMA SELECT USING AR FROM BURG ROUTINE.
C                         AR INVERTED TO LARGE ORDER MA TO PRODUCE
C                         COVARIANCE MATRIX FOR SELECT.
C                     3 - ARMA SELECT USING TRUNCATED INFINITE MA
C                         DERIVED USING CEPSTRAL CORRELATIONS.
C                     4 - BOTH 2 AND 3 PERFORMED
C
C     OUTPUT  VARIOUS MODELS AND DIAGNOSTICS.
C             MODELS. AR BY BEST CAT ORDER
C                     AR BY SECOND BEST CAT ORDER
C                     AR BY BURG ALGORITHM USING LARGER OF BEST AND
C                           SECOND BEST ORDERS
C                     MA BY INVERTING BURG AR
C                     ARMA BY USING COVARIANCES FROM ABOVE MA IN
C                           SELECT REGRESSION ROUTINE
C                     MA BY CEPSTRAL CORRELATIONS
C                     ARMA BY USING COVARIANCES OF CEPSTRAL MA IN
C                           SELECT REGRESSION ROUTINE
C
```

```
C     SUBPROGRAMS CALLED  DTCVF,CVCR,OUENT,DESTAT,PPLOT,KSD,FCODEA,
C                         MINMAX,OUICK,FOFNC,OTOFO,WSPACE,OLOCAL,
C                         PPLOTG,FTERP,MIN,MAX,PREPSP,CUMSP,SPPLT,
C                         PCORR,CVPART,CLPLT1,PARTRV,AICLS,PAR2,
C                         RELMN,CVARP,ARSP,OMEMRY,MAXSPA,CVSPW,DTARB,
C                         PARTAR,MACV,CLPLT2,ARMA,MAMXSL,SELREG,SLMX,
C                         MXSPSL
C
C***********************************************************************
      COMMON /DAT/ YT(1000),CT(1200),ST(1200),FT(1200),WKM(61,61),
     &        IERR(800),R(250),CORR(250),INDT(50),COFT(50),
     &        AIC(250),ALRV(250),
     &        SIG(2),MINVC(2),NAMES(250),
     &        LABY(20),IC1(10),ALPHTP(250)
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION BETA(250),LABPA(20),LABAR2(20)
      DIMENSION LL2(2,2),LABP(20),LABC(20),LABAR1(20),LABW(20)
      DATA LABP/'RAW ','PERI','ODOG','RAM ',16*'    '/
      DATA LABC/'CORR','ELOG','RAM ',17*'    '/
      DATA LABPA/'PART','IAL ','AUTO','CORR','ELAT','IONS',14*'    '/
      DATA LABAR1/'BEST',' ORD','ER A','R SP','ECTR','AL D',
     +           'ENSI','TY  ',12*'    '/
      DATA LABAR2/'SPEC','TRAL',' DEN','SITY',' FRO','M BU','RG A',
     +'R MO','DEL ',11*'    '/
      DATA LABW/'SMOO','THED',' PER','IODO','GRAM',' - P','ARZE',
     +           'N WI','NDOW',11*'    '/
      DATA MDIM/61/
      TWOPI=8.0*ATAN(1.0)
C
C     COMPUTE PERIODOGRAM, CORRELATIONS
C
      CALL DTCVF(YT,NYT,NFREOS,NCOVM,CT,ST,FT,R,RO)
      DIV=RO/TWOPI
      N1=(NFREOS/2)+1
      DELF=TWOPI/FLOAT(NFREOS)
      CALL CVCR(FT,DIV,NFREOS)
C
C     PERFORM ONESAM ANALYSIS ON RAW PERIODOGRAM
C
      CALL OUENT(NFREOS,FT,LABY,LABP,2,1,NFREOS+4,P25,P50,P75)
C
C     LOCAL QUANTILE ANALYSIS OF PERIODOGRAM
C
      IF(NOMP.LE.0) GOTO 605
      NOL=(NFREOS+NOMP)/2+10
      DIVO=1./FLOAT(NFREOS)
      CT(1)=DIVO
      DO 600 I=2,NOL
  600 CT(I)=CT(I-1)+DIVO
      CALL OLOCAL(FT,CT,NOL,NOMP,LABP)
  605 CONTINUE
C
C     PRINT PERIODOGRAM AND CUMULATIVE PERIODOGRAM
C
      CALL PREPSP(FT,NFREOS,1,1.0,CT,SPMIN,SPMAX)
      SPLR=SPMAX-SPMIN
      WRITE(IUNIT,610) SPMIN,SPMAX,RO,SPLR
  610 FORMAT(1H1,10X,26HFOR PERIODOGRAM, SPLMIN = ,F12.5,2X,
     119HSPLMAX = ,F12.5,2X,7HR(O) = ,F12.5,/,T29,'SPLR =',F12.5)
      FFREO=0.0
      CALL SPPLT(FT,N1,FFREO,DELF,1)
      NFRO=NFREOS
      WRITE(IUNIT,620)
  620 FORMAT(//,10X,24HCUMULATIVE PERIODOGRAM :)
      CALL CUMSP(FT,NFRO,1,CT,N1)
C
C     PRINT CORRELATIONS
C
      WRITE(IUNIT,625)
  625 FORMAT(1H1,T10,'TIME DOMAIN ANALYSIS: CORRELOGRAM',//)
      CALL PCORR(R,RO,NCOVM,CORR)
C
C     PERFORM QUANTILE ANALYSIS OF CORRELOGRAM
C
      CALL OUENT(NCOVM,CORR,LABY,LABC,1,0,NCOVM+4,C25,C50,C75)
C
C     CREATE INFORMATION LAG FUNCTION
C
      DO 708 I=1,NCOVM
  708 FT(I)=-0.5*ALOG(1.-CORR(I)*CORR(I))
      WRITE(IUNIT,709)
  709 FORMAT(/,10X,'INFORMATION LAG FUNCTION: -0.5*LOG(1-RHO**2)')
      CALL CLPLT1(FT,NCOVM,1,4HINFO,41,1)
      WRITE(IUNIT,710)
  710 FORMAT(1H1,/,10X,27HAR DESCRIPTION OF Y TILDA :,//,10X,
     +'PARTIAL AUTOCORRELATIONS VIA THE YULE WALKER EOUATIONS:')
      CALL CVPART(R,RO,NCOVM,CT,ST,FT)
C
C     FT CONTAINS PARTIAL AUTOCORRELATIONS
C     PRINT PARTIAL AUTOCORRELATIONS
C
      CALL CLPLTD(FT,NCOVM,1,4HPACF,1.0,-1.0)
C
C     PERFORM QUANTILE ANALYSIS OF PARTIAL AUTOCORRELATIONS
C
      CALL OUENT(NCOVM,FT,LABY,LABPA,1,0,NCOVM+4,C25,C50,C75)
C
C     COMPUTE PREDICTION VARIANCES AND AIC
C
      CALL PARTRV(FT,NCOVM,1.0,ST)
      CALL AICLS(ST,NCOVM,NYT,ALRV,AIC,LAIC)
      WRITE(IUNIT,961) LAIC
  961 FORMAT(//,10X,15HORDER BY AIC = ,I3)
      CALL CLPLT2(ALRV,AIC,NCOVM,1,4HALRV,4H AIC,81,1)
C
C     COMPUTE CAT AR ORDERS
C
      DEC=1./FLOAT(NYT)
      CALL PAR2(ST,NCOVM,NYT,FT,NORD)
      WRITE(IUNIT,970)
  970 FORMAT(///)
      CALL RELMN(FT,NCOVM,DEC,MIN1,MIN2)
      IF(FT(MIN1).LE.-1.) GOTO 731
      WRITE(IUNIT,1401)
 1401 FORMAT(//,10X,'ORDER BY CAT = 0.')
  731 CONTINUE
C
C     AR SPECTRAL ANALYSIS OF Y
C
      NORDTP=MIN1
      IF((MIN1+MIN2).EO.0) GO TO 785
      RVTP=ST(MIN1)
      IF(MIN2.EO.0) MIN2=MIN1
      SIG(1)=ST(MIN1)
      SIG(2)=ST(MIN2)
      MINVC(1)=MIN1
      MINVC(2)=MIN2
      DO 780 IORD=1,2
      IF((IORD.EO.2).AND.(MIN1.EO.MIN2)) GO TO 785
      WRITE(IUNIT,725) MINVC(IORD),SIG(IORD)
  725 FORMAT(/,10X,'FOR ORDER ',I2,' AR MODEL, RVAR =',F10.7)
      WRITE(IUNIT,730) LL2(1,IORD),LL2(2,IORD)
```

```
  730 FORMAT(//,10X,17HCOEFFICIENTS FOR ,2A4,1X,7HORDER :)                    ARSPIO
C                                                                             ARSPIO
C     COMPUTE AR COEFFICIENTS FROM CORRELATIONS                               ARSPIO
C                                                                             ARSPIO
      CALL CVARP(CORR,NCOVM,MINVC(IORD),ST,ALPHTP)                            ARSPIO
      CALL CLPLT1(ALPHTP,MINVC(IORD),1,4HALPH,41,1)                           ARSPIO
      IF(IORD.EQ.2) GO TO 780                                                 ARSPIO
      SIG(IORD)=SIG(IORD)*TWOPI                                               ARSPIO
C                                                                             ARSPIO
C     COMPUTE AR SPECTRAL DENSITY                                             ARSPIO
C                                                                             ARSPIO
      CALL ARSP(ALPHTP,SIG(IORD),MINVC(IORD),NFREQS,ST,FT)                    ARSPIO
C                                                                             ARSPIO
C     CREATE DELTA SEQUENCE FOR AR SPECTRAL DENSITY                           ARSPIO
C                                                                             ARSPIO
      CALL DMEMRY(FT,NFREQS,XSEAS,CT,ST,LABAR1)                               ARSPIO
      CALL PREPSP(FT,NFREQS,1,1.0,ST,SPMIN,SPMAX)                             ARSPIO
      SPLR=SPMAX-SPMIN                                                        ARSPIO
      WRITE(IUNIT,760) SPMIN,SPMAX,SPLR                                       ARSPIO
  760 FORMAT(//,10X,18HSPECTRA, SPLMIN = ,F12.5,2X,                           ARSPIO
     19HSPLMAX = ,F12.5,4X,'SPLR = ',F12.6)                                   ARSPIO
      CALL SPPLT(FT,N1,FFREQ,DELF,1)                                          ARSPIO
      CALL MAXSPA(FT,ALPHTP,NFREQS,MINVC(IORD),100,1.E-6,NOFM,ST,IERR)        ARSPIO
      NFRQ=NFREQS                                                             ARSPIO
      WRITE(IUNIT,770) LL2(1,IORD),LL2(2,IORD)                                ARSPIO
  770 FORMAT(//,10X,11HCUMULATIVE ,2A4,1X,15HORDER SPECTRA :)                 ARSPIO
      CALL CUMSP(FT,NFRQ,1,CT,N1)                                             ARSPIO
  780 CONTINUE                                                                ARSPIO
  785 CONTINUE                                                                ARSPIO
C                                                                             ARSPIO
C     COMPUTE SMOOTHED PERIODOGRAM USIN  THE PARZEN WINDOW.                   ARSPIO
C                                                                             ARSPIO
      NTRUNC=(NYT/4)*2                                                        ARSPIO
      IF(NTRUNC.GT.NCOVM) NTRUNC=(NCOVM/2)*2                                  ARSPIO
      IOPTW=6                                                                 ARSPIO
      CALL CVSPW(R,RO,IOPTW,NTRUNC,NFREQS,CT,ST,FT)                           ARSPIO
      CALL CVCR(FT,DIV,NFREQS)                                                ARSPIO
      WRITE(IUNIT,784) NTRUNC                                                 ARSPIO
  784 FORMAT(/,10X,'TRUNCATION POINT FOR SMOOTHED PERIODOGRAM = ',I3)         ARSPIO
C                                                                             ARSPIO
C     CREATE DELTA SEQUENCE FOR SMOOTHED PERIODOGRAM                          ARSPIO
C                                                                             ARSPIO
      CALL DMEMRY(FT,NFREQS,XSEAS,CT,ST,LABW)                                 ARSPIO
C                                                                             ARSPIO
C     MIXED SCHEME SELECT PROCEDURE                                           ARSPIO
C                                                                             ARSPIO
C             IOPTMX.   1 - NO ARMA SELECT MODELING                           ARSPIO
C                       2 - ARMA SELECT USING AR FROM BURG ROUTINE.           ARSPIO
C                           AR INVERTED TO LARGE ORDER MA TO PRODUCE          ARSPIO
C                           COVARIANCE MATRIX FOR SELECT                      ARSPIO
C                       3 - ARMA SELECT USING TRUNCATED INFINITE MA           ARSPIO
C                           DERIVED USING CEPSTRAL CORRELATIONS.              ARSPIO
C                       4 - BOTH 2 AND 3 PERFORMED.                           ARSPIO
C                                                                             ARSPIO
      IF(IOPTMX.EQ.1) GO TO 885                                               ARSPIO
      ICHK=0                                                                  ARSPIO
      WRITE(IUNIT,800)                                                        ARSPIO
  800 FORMAT(1H1,/,10X,29(1H*),/,10X,'MIXED SCHEME SELECT PROCEDURE',         ARSPIO
     */,10X,29(1H*),//)                                                       ARSPIO
      IF(IOPTMX.GE.3) GO TO 810                                               ARSPIO
      ICHK=ICHK+1                                                             ARSPIO
      GO TO 835                                                               ARSPIO
C                                                                             ARSPIO
C     COMPUTE BETAS OF TRUNCATED INFINITE MA FROM CEPSTRAL                    ARSPIO
C     CORRELATIONS                                                            ARSPIO
C                                                                             ARSPIO
  810 CALL CPMA(FT,NFREQS,NCOVM,CT,ST,BETA,RVMA,SIGINF)                       ARSPIO
      GO TO 849                                                               ARSPIO
  835 ITRUNC=MINVC(1)                                                         ARSPIO
      IF(ITRUNC.LT.MINVC(2)) ITRUNC=MINVC(2)                                  ARSPIO
      IF(ITRUNC.EQ.0) GO TO 865                                               ARSPIO
C                                                                             ARSPIO
C     COMPUTE PARTIAL AUTOCORRELATIONS USING THE BURG ALGORITHM               ARSPIO
C     THEN PRODUCE AR COEFFICIENTS FOR ORDER ITRUNC.                          ARSPIO
C                                                                             ARSPIO
      CALL DTARB(YT,NYT,ITRUNC,2,CT,ST,FT,RO,AIC,ALRV)                        ARSPIO
      WRITE(IUNIT,838) RO                                                     ARSPIO
  838 FORMAT(/,10X,'FROM DTARB, RO=',F15.5)                                   ARSPIO
C                                                                             ARSPIO
C     AIC CONTAINS PAC.F., ALRV CONTAINS RESIDUAL VARIANCE.                   ARSPIO
C     PLOT PARTIAL AUTOCORRELATIONS.                                          ARSPIO
C                                                                             ARSPIO
      RVMA=ALRV(ITRUNC)/RO                                                    ARSPIO
      WRITE(IUNIT,836)                                                        ARSPIO
  836 FORMAT(/,10X,'PARTIAL AUTOCORRELATIONS VIA BURGS ALGORITHM:',/)         ARSPIO
      CALL CLPLTD(AIC,ITRUNC,1,4HPACF,1.0,-1.0)                               ARSPIO
      CALL PARTAR(AIC,ITRUNC,ALPHTP)                                          ARSPIO
      CALL MACV(ALPHTP,1.0,ITRUNC,R,RO)                                       ARSPIO
      CALL CVCR(R,RO,ITRUNC)                                                  ARSPIO
      WRITE(IUNIT,837) RO                                                     ARSPIO
  837 FORMAT(//,10X,38HCOEFFICIENTS AND INVERSE CORRELATIONS ,                ARSPIO
     *//,10X,33HSUM OF SQUARES OF COEFFICIENTS = ,F10.4)                      ARSPIO
      CALL CLPLT2(ALPHTP,R,ITRUNC,1,4HALPH,4HCORI,81,1)                       ARSPIO
      RVMAP=RVMA*TWOPI                                                        ARSPIO
      CALL ARSP(ALPHTP,RVMAP,ITRUNC,NFREQS,ST,FT)                             ARSPIO
      CALL DMEMRY(FT,NFREQS,XSEAS,CT,ST,LABAR2)                               ARSPIO
      CALL PREPSP(FT,NFREQS,1,1.0,ST,SPMIN,SPMAX)                             ARSPIO
      SPLR=SPMAX-SPMIN                                                        ARSPIO
      WRITE(IUNIT,760) SPMIN,SPMAX,SPLR                                       ARSPIO
      CALL SPPLT(FT,N1,FFREQ,DELF,1)                                          ARSPIO
      CALL MAXSPA(FT,ALPHTP,NFREQS,ITRUNC,100,1.E-6,NOFM,ST,IERR)             ARSPIO
      NFRQ=NFREQS                                                             ARSPIO
      WRITE(IUNIT,842)                                                        ARSPIO
  842 FORMAT(//,10X,'CUMULATIVE SPECTRA FOR BURG AR:')                        ARSPIO
      CALL CUMSP(FT,NFRQ,1,CT,N1)                                             ARSPIO
      CALL ARMA(ALPHTP,ITRUNC,NCOVM,BETA)                                     ARSPIO
      WRITE(IUNIT,848) RVMA                                                   ARSPIO
  848 FORMAT(//,10X,'MOVING AVERAGE MODEL VIA SUBR. ARMA. RVAR=',             ARSPIO
     *F10.5,//,10X,'FIRST 10 COEFFICIENTS OF INFINITE MA:',//)                ARSPIO
      CALL CLPLT1(BETA,10,1,4HBETA,41,1)                                      ARSPIO
  849 ICHK=ICHK+1                                                             ARSPIO
C                                                                             ARSPIO
C     DETERMINE MAX AR AND MA ORDERS FOR SELECT                               ARSPIO
C                                                                             ARSPIO
      K1=NCOVM/2-1                                                            ARSPIO
      IF(K1.GT.30) K1=30                                                      ARSPIO
      K2=K1                                                                   ARSPIO
      CALL MACV(BETA,RVMA,NCOVM,R,RO)                                         ARSPIO
      CALL CVCR(R,RO,NCOVM)                                                   ARSPIO
      CALL MAM  L(R,1.0,BETA,NCOVM,MDIM,K1,K2,0,NYT,WKM,                      ARSPIO
     *KPS,NAM S,COFT,INDT,RVMA,IER)                                           ARSPIO
      IRV=IABS(K1)+IABS(K2)+1                                                 ARSPIO
       VS=WK  IRV  RV)                                                        ARSPIO
      C LL S   ,INDT,COFT,ALPHTP,BETA,NORDAR,NORDMA)                          ARSPIO
      WRITE(  NIT,852)                                                        ARSPIO
  852 FORMAT(//,10X,'TO INTERPRET CRITICAL ADD AND DELETE VALUES, ',          ARSPIO
     * NOT  TH   ,/,10X,'THEY ARE COMPUTED USING TWICE THE SAMPLE',           ARSPIO
     *  SIZE  )                                                               ARSPIO
      WRITE(IUNIT,850) RVS                                                    ARSPIO
  850 F    (//,10X,37HRESIDUAL VARIANCE FROM SUBSET ARMA = ,F10.7)            ARSPIO
      I (NORDAR.EQ.0.AND.NORDMA.EQ.0) GO TO 880                               ARSPIO
       NORDAR.EQ.0) GO TO 855                                                 ARSPIO
```

```
      CALL CLPLT1(ALPHTP,NORDAR,1,4HALPH,61,1)
  855 IF(NORDMA.EQ.0) GO TO 860
      CALL CLPLT1(BETA,NORDMA,1,4HBETA,61,1)
  860 CONTINUE
      RVS1=RVS*TWOPI
      CALL MXSPSL(COPT,INDT,RVS1,NPS,NFREQS,CT,ST,FT)
      CALL PREPSP(FT,NFREQS,1,1,0,ST,SPMIN,SPMAX)
      SPLR=SPMAX-SPMIN
      WRITE(IUNIT,760) SPMIN,SPMAX,SPLR
      CALL SPPLT(FT,N1,FFREQ,DELF,1)
      NFRQ=NFREQS
      WRITE(IUNIT,875)
  875 FORMAT(//,10X,32HCUMULATIVE SUBSET ARMA SPECTRA   )
      CALL CUMSP(FT,NFRQ,1,CT,N1)
  880 IF(ICHK.EQ.1.AND.IOPTMX.NE.3) GO TO 835
  885 CONTINUE
  999 RETURN
      END
      SUBROUTINE DMEMRY(FT,NFREQS,XSEAS,CT,ST,LABEL)
C**************************************************************
C
C    SUBROUTINE TO COMPUTE DELTA MEMORY SEQUENCE FOR SPECTRAL
C    DENSITY FUNCTION FT
C
C    INPUT: FT - SPECTRAL DENSITY
C           NFREQS - NUMBER OF EQUALLY SPACED FREQUENCIES USED IN
C                    COMPUTING FT
C           XSEAS - SEASONAL LAG, I.E., XSEAS=12 FOR MONTHLY DATA, ETC
C           LABEL - LABEL GIVING TYPE OF INPUT SPECTRAL DENSITY
C
C    AUXILLIARY: CT,ST
C
C    OUTPUT: PRINTER PLOT OF DELTA SEQUENCE
C
C    SUBPROGRAMS CALLED: CLPLT1
C
C**************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION FT(1),CT(1),ST(1),LABEL(20)
      XSNF=1./XSEAS
      WRITE(IUNIT,100) LABEL,XSNF
      KFREQ=XSNF*FLOAT(NFREQS)
      INIT=0
      KFREQ1=KFREQ+1
   10 CONTINUE
      CT(1)=ALOG(FT(INIT+1))
      ST(1)=CT(1)
      DO  20 I=2,KFREQ1
      CT(I)=ALOG(FT(I+INIT))
   20 ST(I)=ST(I-1)+CT(I)
      DO  30 I=1,KFREQ
      ST(I)=ST(I)/FLOAT(I)-CT(I+1)
      IF(ST(I).GT.3.0) ST(I)=3.0
      IF(ST(I).LT.-3.0) ST(I)=-3.0
   30 CONTINUE
      CALL CLPLTD(ST,KFREQ,1,4HDELT,3.0,-3.0)
      IF(INIT.NE.0) GO TO 40
      INIT=KFREQ
      GO TO 10
   40 RETURN
  100 FORMAT(1H1,9X,'DELTA MEMORY FUNCTION',//,10X,20A4,//10X,
     +'PLOT 1 - LAG 1 IS AT FREQUENCY 0',/10X,
     +'PLOT 2 - LAG 1 IS AT FREQUENCY ',F7.5,//)
      END
      FUNCTION DOT(X,Y,N)
C**********************************************************
C
C    INNER PRODUCT OF TWO VECTORS.
C
C**********************************************************
      DIMENSION X(1),Y(1)
      DOUBLE PRECISION C
      C=0.0D0
      DO 10 I=1,N
   10 C=C+DBLE(X(I))*DBLE(Y(I))
      DOT=C
      RETURN
      END
      SUBROUTINE DTARB(DAT,N,NP,IOPT,Y,X,YP,R0,AJ,RVAR)
C**************************************************************
C
C    BURGS ALGORITHM (IOPT=2) OR TOEPLITZ GRAM-SCHMIDT ALGORITHM
C    FOR ESTIMATING PARTIAL AUTOCORRELATIONS AJ(1),...,AJ(NP) AND
C    RESIDUAL VARIANCES RVAR(1),...,RVAR(NP) FOR AR ORDERS 1 THRU NP.
C
C**************************************************************
      DIMENSION DAT(1),Y(1),X(1),YP(1),AJ(1),RVAR(1)
      NPNP=N+NP
      DO 1 I=1,NPNP
      Y(I)=0.0
      X(I)=0.0
    1 YP(I)=0.0
      DO 2 I=1,N
      Y(I)=DAT(I)
    2 X(I+1)=DAT(I)
      R0=DOT(DAT,DAT,N)/FLOAT(N)
      SIG=R0
      DO 100 J=1,NP
      IF(IOPT.EQ.2) GO TO 10
      AJ(J)=-2.*DOT(X,Y,NPNP)/(DOT(X,X,NPNP)+DOT(Y,Y,NPNP))
      GO TO 20
   10 TOP=-2.*DOT(X(J+1),Y(J+1),N-J)
      BOT=DOT(X(J+1),X(J+1),N-J)+DOT(Y(J+1),Y(J+1),N-J)
      AJ(J)=TOP/BOT
   20 CONTINUE
      AJJ=AJ(J)
      SIG=SIG*(1.-AJJ*AJJ)
      RVAR(J)=SIG
      DO 30 K=1,NPNP
      YP(K)=Y(K)
      Y(K)=Y(K)+AJJ*X(K)
   30 X(K)=X(K)+AJJ*YP(K)
      DO 40 K=2,NPNP
      KK=NPNP-K+2
   40 X(KK)=X(KK-1)
      X(1)=0.0
  100 CONTINUE
      RETURN
      END
      SUBROUTINE DTCVF(X,N,NFREQS,M,CT,ST,PER,R,R0)
C**********************************************************
C
C    SUBROUTINE TO CALCULATE THE FIRST M+1 SAMPLE AUTO-
C    COVARIANCES R0,R(1),...,R(M) OF A SAMPLE X(1),...,X(N)
C    FROM A TIME SERIES X(.) VIA THE FAST FOURIER TRANSFORM.
C
C    INPUT :
C             N,X(1),...,X(N),M
C             NFREQS : AN INTEGER (>N+M, LARGEST PRIME
C             FACTOR <23) SPECIFYING THE NUMBER OF EQUALLY
C             SPACED FREQUENCIES BETWEEN 0 AND TWOPI AT
C             WHICH THE SAMPLE SPECTRAL DENSITY (PER) IS
```

```
C              TO BE CALCULATED
C
C     OUTPUT
C              PER(1),...,PER(NFREQS),R0,R(1),...,R(M)
C
C     AUXILLIARY
C              CT,ST
C
C     SUBROUTINES CALLED    DTSP SPCV FFT
C
C*****************************************************************
C
      DIMENSION X(N),CT(NFREQS),ST(NFREQS),PER(NFREQS),R(M)
C
      CALL DTSP(X,N,NFREQS,CT,ST,PER)
      CALL SPCV(PER,NFREQS,M,CT,ST,R,R0)
C
      RETURN
      END
      SUBROUTINE DTSP(X,N,NFREQS,CT,ST,PER)
C*****************************************************************
C
C     SUBROUTINE TO CALCULATE THE SAMPLE SPECTRAL DENSITYPER
C     AT THE NFREQS EQUALLY SPACED FREQUENCIES BETWEEN 0
C     AND TWOPI OF THE SAMPLE X(1),...,X(N), N < NFREQS, FROM
C     A TIME SERIES X
C
C     INPUT
C              N,X(1),...,X(N)
C              NFREQS ( LARGEST PRIME FACTOR < 23)
C
C     OUTPUT
C              PER(1),...,PER(NFREQS)
C
C     AUXILLIARY
C              CT,ST
C
C     SUBROUTINES CALLED    FFT
C
C*****************************************************************
C
      DIMENSION X(N),CT(NFREQS),ST(NFREQS),PER(NFREQS)
C
      TWOPIN=8.*ATAN(1.0)*FLOAT(N)
C
      DO 1 I=1,NFREQS
      CT(I)=0
    1 ST(I)=0
      DO 2 I=1,N
    2 CT(I)=X(I)
C
      CALL FFT(CT,ST,NFREQS,NFREQS,NFREQS,1)
C
      DO 3 I=1,NFREQS
    3 PER(I)=(CT(I)*CT(I)+ST(I)*ST(I))/TWOPIN
C
      RETURN
      END
      SUBROUTINE FCODEA(X,IFORM,NAME)
C*****************************************************************
C     SUBROUTINE TO CONVERT REAL VARIABLE X
C     WHICH HAS 4 CHARACTER F-FORMAT IFORM
C     TO 8 CHARACTER ALPHAMERIC ARRAY NAME WHICH IS
C     IN A FORMAT.
C     INPUT  : NSCRCH : SCRATCH TAPE NUMBER
C              X, IFORM
C     OUTPUT : NAME(1),NAME(2) . 4 CHARACTERS EACH
C*****************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION NAME(2),IFORM(2)
      REWIND NSCRCH
      WRITE(NSCRCH,IFORM)X
      REWIND NSCRCH
      READ(NSCRCH,10)NAME
   10 FORMAT(2A4)
      RETURN
      END
      SUBROUTINE FFT(A,B,NTOT,N,NSPAN,ISN)
C  MULTIVARIATE COMPLEX FOURIER TRANSFORM, COMPUTED IN PLACE
C    USING MIXED-RADIX FAST FOURIER TRANSFORM ALGORITHM.
C  BY R. C. SINGLETON, STANFORD RESEARCH INSTITUTE, OCT. 1968
C  ARRAYS A AND B ORIGINALLY HOLD THE REAL AND IMAGINARY
C    COMPONENTS OF THE DATA, AND RETURN THE REAL AND
C    IMAGINARY COMPONENTS OF THE RESULTING FOURIER COEFFICIENTS.
C  MULTIVARIATE DATA IS INDEXED ACCORDING TO THE FORTRAN
C    ARRAY ELEMENT SUCCESSOR FUNCTION, WITHOUT LIMIT
C    ON THE NUMBER OF IMPLIED MULTIPLE SUBSCRIPTS.
C    THE SUBROUTINE IS CALLED ONCE FOR EACH VARIATE.
C    THE CALLS FOR A MULTIVARIATE TRANSFORM MAY BE IN ANY ORDER.
C  NTOT IS THE TOTAL NUMBER OF COMPLEX DATA VALUES.
C  N IS THE DIMENSION OF THE CURRENT VARIABLE
C  NSPAN/N IS THE SPACING OF CONSECUTIVE DATA VALUES
C    WHILE INDEXING THE CURRENT VARIABLE.
C  THE SIGN OF ISN DETERMINES THE SIGN OF THE COMPLEX
C    EXPONENTIAL, AND THE MAGNITUDE OF ISN IS NORMALLY ONE.
C   NOTE  WHEN TRANSFORMING DATA TO THE FREQUENCY DOMAIN(ISN=+1),
C      FFT YIELDS THE REAL AND IMAGINARY PARTS OF Z(FREQ), WHERE
C      Z(FREQ)=SUM(J=1 TO N) OF X(J)*EXP(I*(J-1)*FREQ), WHERE FREQ
C      REPRESENTS ONE OF THE N FREQUENCIES ON THE GRID 0 TO
C      (2*PI*(N-1)/N)).
C   ALSO NOTE  WHEN TRANSFORMING FOURIER COEFFICIENTS BACK TO
C      THE TIME DOMAIN(ISN=-1), FFT YIELDS THE REAL AND IMAGINARY
C      PARTS OF N*X(J), FOR J=1 TO N.
C  A TRI-VARIATE TRANSFORM WITH A(N1,N2,N3), B(N1,N2,N3)
C    IS COMPUTED BY
C      CALL FFT(A,B,N1*N2*N3,N1,N1,1)
C      CALL FFT(A,B,N1*N2*N3,N2,N1*N2,1)
C      CALL FFT(A,B,N1*N2*N3,N3,N1*N2*N3,1)
C  FOR A SINGLE-VARIATE TRANSFORM,
C    NTOT = N = NSPAN = (NUMBER OF COMPLEX DATA VALUES), E.G.
C      CALL FFT(A,B,N,N,N,1)
C  THE DATA MAY ALTERNATIVELY BE STORED IN A SINGLE COMPLEX
C    ARRAY A, THEN THE MAGNITUDE OF ISN CHANGED TO TWO TO
C    GIVE THE CORRECT INDEXING INCREMENT AND A(2) USED TO
C    PASS THE INITIAL ADDRESS FOR THE SEQUENCE OF IMAGINARY
C    VALUES, E.G.
C      CALL FFT(A,A(2),NTOT,N,NSPAN,2)
C  ARRAYS AT(MAXF), CK(MAXF), BT(MAXF), SK(MAXF), AND NP(MAXP)
C    ARE USED FOR TEMPORARY STORAGE.  IF THE AVAILABLE STORAGE
C    IS INSUFFICIENT, THE PROGRAM IS TERMINATED BY A STOP.
C    MAXF MUST BE .GE. THE MAXIMUM PRIME FACTOR OF N.
C    MAXP MUST BE .GT. THE NUMBER OF PRIME FACTORS OF N.
C    IN ADDITION, IF THE SQUARE-FREE PORTION K OF N HAS TWO OR
C    MORE PRIME FACTORS, THEN MAXP MUST BE .GE. K-1.
      DIMENSION A(1),B(1)
C  ARRAY STORAGE IN NFAC FOR A MAXIMUM OF 11 FACTORS OF N.
C  IF N HAS MORE THAN ONE SQUARE-FREE FACTOR, THE PRODUCT OF THE
C    SQUARE-FREE FACTORS MUST BE .LE. 210
      DIMENSION NFAC(11),NP(209)
C  ARRAY STORAGE FOR MAXIMUM PRIME FACTOR OF 23
      DIMENSION AT(23),CK(23),BT(23),SK(23)
      EQUIVALENCE (I,II)
```

```
C  THE FOLLOWING TWO CONSTANTS SHOULD AGREE WITH THE ARRAY DIMENSIONS.
      MAXF=23
      MAXP=209
      IF(N .LT. 2) RETURN
      RAD=8.0*ATAN(1.0)
      S72=RAD/5.0
      C72=COS(S72)
      S72=SIN(S72)
      S120=SQRT(0.75)
      INC=ISN
      IF(ISN .GE. 0) GO TO 10
      S72=-S72
      S120=-S120
      RAD=-RAD
      INC=-INC
   10 NT=INC*NTOT
      KS=INC*NSPAN
      KSPAN=KS
      NN=NT-INC
      JC=KS/N
      RADF=RAD*FLOAT(JC)*0.5
      I=0
      JF=0
C  DETERMINE THE FACTORS OF N
      M=0
      K=N
      GO TO 20
   15 M=M+1
      NFAC(M)=4
      K=K/16
   20 IF(K-(K/16)*16 .EQ. 0) GO TO 15
      J=3
      JJ=9
      GO TO 30
   25 M=M+1
      NFAC(M)=J
      K=K/JJ
   30 IF(MOD(K,JJ) .EQ. 0) GO TO 25
      J=J+2
      JJ=J**2
      IF(JJ .LE. K) GO TO 30
      IF(K .GT. 4) GO TO 40
      KT=M
      NFAC(M+1)=K
      IF(K .NE. 1) M=M+1
      GO TO 80
   40 IF(K-(K/4)*4 .NE. 0) GO TO 50
      M=M+1
      NFAC(M)=2
      K=K/4
   50 KT=M
      J=2
   60 IF(MOD(K,J) .NE. 0) GO TO 70
      M=M+1
      NFAC(M)=J
      K=K/J
   70 J=((J+1)/2)*2+1
      IF(J .LE. K) GO TO 60
   80 IF(KT .EQ. 0) GO TO 100
      J=KT
   90 M=M+1
      NFAC(M)=NFAC(J)
      J=J-1
      IF(J .NE. 0) GO TO 90
C  COMPUTE FOURIER TRANSFORM
  100 SD=RADF/FLOAT(KSPAN)
      CD=2.0*SIN(SD)**2
      SD=SIN(SD+SD)
      KK=1
      I=I+1
      IF(NFAC(I) .NE. 2) GO TO 400
C  TRANSFORM FOR FACTOR OF 2 (INCLUDING ROTATION FACTOR)
      KSPAN=KSPAN/2
      K1=KSPAN+2
  210 K2=KK+KSPAN
      AK=A(K2)
      BK=B(K2)
      A(K2)=A(KK)-AK
      B(K2)=B(KK)-BK
      A(KK)=A(KK)+AK
      B(KK)=B(KK)+BK
      KK=K2+KSPAN
      IF(KK .LE. NN) GO TO 210
      KK=KK-NN
      IF(KK .LE. JC) GO TO 210
      IF(KK .GT. KSPAN) GO TO 800
  220 C1=1.0-CD
      S1=SD
  230 K2=KK+KSPAN
      AK=A(KK)-A(K2)
      BK=B(KK)-B(K2)
      A(KK)=A(KK)+A(K2)
      B(KK)=B(KK)+B(K2)
      A(K2)=C1*AK-S1*BK
      B(K2)=S1*AK+C1*BK
      KK=K2+KSPAN
      IF(KK .LT. NT) GO TO 230
      K2=KK-NT
      C1=-C1
      KK=K1-K2
      IF(KK .GT. K2) GO TO 230
      AK=C1-(CD*C1+SD*S1)
      S1=(SD*C1-CD*S1)+S1
      C1=2.0-(AK**2+S1**2)
      S1=C1*S1
      C1=C1*AK
      KK=KK+JC
      IF(KK .LT. K2) GO TO 230
      K1=K1+INC+INC
      KK=(K1-KSPAN)/2+JC
      IF(KK .LE. JC+JC) GO TO 220
      GO TO 100
C  TRANSFORM FOR FACTOR OF 3 (OPTIONAL CODE)
  320 K1=KK+KSPAN
      K2=K1+KSPAN
      AK=A(KK)
      BK=B(KK)
      AJ=A(K1)+A(K2)
      BJ=B(K1)+B(K2)
      A(KK)=AK+AJ
      B(KK)=BK+BJ
      AK=-0.5*AJ+AK
      BK=-0.5*BJ+BK
      AJ=(A(K1)-A(K2))*S120
      BJ=(B(K1)-B(K2))*S120
      A(K1)=AK-BJ
      B(K1)=BK+AJ
      A(K2)=AK+BJ
      B(K2)=BK-AJ
      KK=K2+KSPAN
      IF(KK .LT. NN) GO TO 320
      KK=KK-NN
```

```
      IF(KK .LE. KSPAN) GO TO 320
      GO TO 700
C  TRANSFORM FOR FACTOR OF 4
  400 IF(NFAC(I) .NE. 4) GO TO 600
      KSPNN=KSPAN
      KSPAN=KSPAN/4
  410 C1=1.0
      S1=0
  420 K1=KK+KSPAN
      K2=K1+KSPAN
      K3=K2+KSPAN
      AKP=A(KK)+A(K2)
      AKM=A(KK)-A(K2)
      AJP=A(K1)+A(K3)
      AJM=A(K1)-A(K3)
      A(KK)=AKP+AJP
      AJP=AKP-AJP
      BKP=B(KK)+B(K2)
      BKM=B(KK)-B(K2)
      BJP=B(K1)+B(K3)
      BJM=B(K1)-B(K3)
      B(KK)=BKP+BJP
      BJP=BKP-BJP
      IF(ISN .LT. 0) GO TO 450
      AKP=AKM-BJM
      AKM=AKM+BJM
      BKP=BKM+AJM
      BKM=BKM-AJM
      IF(S1 .EQ. 0.0) GO TO 460
  430 A(K1)=AKP*C1-BKP*S1
      B(K1)=AKP*S1+BKP*C1
      A(K2)=AJP*C2-BJP*S2
      B(K2)=AJP*S2+BJP*C2
      A(K3)=AKM*C3-BKM*S3
      B(K3)=AKM*S3+BKM*C3
      KK=K3+KSPAN
      IF(KK .LE. NT) GO TO 420
  440 C2=C1-(CD*C1+SD*S1)
      S1=(SD*C1-CD*S1)+S1
      C1=2.0-(C2**2+S1**2)
      S1=C1*S1
      C1=C1*C2
      C2=C1**2-S1**2
      S2=2.0*C1*S1
      C3=C2*C1-S2*S1
      S3=C2*S1+S2*C1
      KK=KK-NT+JC
      IF(KK .LE. KSPAN) GO TO 420
      KK=KK-KSPAN+INC
      IF(KK .LE. JC) GO TO 410
      IF(KSPAN .EQ. JC) GO TO 800
      GO TO 100
  450 AKP=AKM+BJM
      AKM=AKM-BJM
      BKP=BKM-AJM
      BKM=BKM+AJM
      IF(S1 .NE. 0.0) GO TO 430
  460 A(K1)=AKP
      B(K1)=BKP
      A(K2)=AJP
      B(K2)=BJP
      A(K3)=AKM
      B(K3)=BKM
      KK=K3+KSPAN
      IF(KK .LE. NT) GO TO 420
      GO TO 440
C  TRANSFORM FOR FACTOR OF 5 (OPTIONAL CODE)
  510 C2=C72**2-S72**2
      S2=2.0*C72*S72
  520 K1=KK+KSPAN
      K2=K1+KSPAN
      K3=K2+KSPAN
      K4=K3+KSPAN
      AKP=A(K1)+A(K4)
      AKM=A(K1)-A(K4)
      BKP=B(K1)+B(K4)
      BKM=B(K1)-B(K4)
      AJP=A(K2)+A(K3)
      AJM=A(K2)-A(K3)
      BJP=B(K2)+B(K3)
      BJM=B(K2)-B(K3)
      AA=A(KK)
      BB=B(KK)
      A(KK)=AA+AKP+AJP
      B(KK)=BB+BKP+BJP
      AK=AKP*C72+AJP*C2+AA
      BK=BKP*C72+BJP*C2+BB
      AJ=AKM*S72+AJM*S2
      BJ=BKM*S72+BJM*S2
      A(K1)=AK-BJ
      A(K4)=AK+BJ
      B(K1)=BK+AJ
      B(K4)=BK-AJ
      AK=AKP*C2+AJP*C72+AA
      BK=BKP*C2+BJP*C72+BB
      AJ=AKM*S2-AJM*S72
      BJ=BKM*S2-BJM*S72
      A(K2)=AK-BJ
      A(K3)=AK+BJ
      B(K2)=BK+AJ
      B(K3)=BK-AJ
      KK=K4+KSPAN
      IF(KK .LT. NN) GO TO 520
      KK=KK-NN
      IF(KK .LE. KSPAN) GO TO 520
      GO TO 700
C  TRANSFORM FOR ODD FACTORS
  600 K=NFAC(I)
      KSPNN=KSPAN
      KSPAN=KSPAN/K
      IF(K .EQ. 3) GO TO 320
      IF(K .EQ. 5) GO TO 510
      IF(K .EQ. JF) GO TO 640
      JF=K
      S1=RAD/FLOAT(K)
      C1=COS(S1)
      S1=SIN(S1)
      IF(JF .GT. MAXF) GO TO 998
      CK(JF)=1.0
      SK(JF)=0.0
      J=1
  630 CK(J)=CK(K)*C1+SK(K)*S1
      SK(J)=CK(K)*S1-SK(K)*C1
      K=K-1
      CK(K)=CK(J)
      SK(K)=-SK(J)
      J=J+1
      IF(J .LT. K) GO TO 630
  640 K1=KK
      K2=KK+KSPNN
      AA=A(KK)
      BB=B(KK)
```

```
      AK=AA
      BK=BB
      J=1
      K1=K1+KSPAN
  650 K2=K2-KSPAN
      J=J+1
      AT(J)=A(K1)+A(K2)
      AK=AT(J)+AK
      BT(J)=B(K1)+B(K2)
      BK=BT(J)+BK
      J=J+1
      AT(J)=A(K1)-A(K2)
      BT(J)=B(K1)-B(K2)
      K1=K1+KSPAN
      IF(K1 .LT. K2) GO TO 650
      A(KK)=AK
      B(KK)=BK
      K1=KK
      K2=KK+KSPNN
      J=1
  660 K1=K1+KSPAN
      K2=K2-KSPAN
      JJ=J
      AK=AA
      BK=BB
      AJ=0.0
      BJ=0.0
      K=1
  670 K=K+1
      AK=AT(K)*CK(JJ)+AK
      BK=BT(K)*CK(JJ)+BK
      K=K+1
      AJ=AT(K)*SK(JJ)+AJ
      BJ=BT(K)*SK(JJ)+BJ
      JJ=JJ+J
      IF(JJ .GT. JF) JJ=JJ-JF
      IF(K .LT. JF) GO TO 670
      K=JF-J
      A(K1)=AK-BJ
      B(K1)=BK+AJ
      A(K2)=AK+BJ
      B(K2)=BK-AJ
      J=J+1
      IF(J .LT. K) GO TO 660
      KK=KK+KSPNN
      IF(KK .LE. NN) GO TO 640
      KK=KK-NN
      IF(KK .LE. KSPAN) GO TO 640
C  MULTIPLY BY ROTATION FACTOR (EXCEPT FOR FACTORS OF 2 AND 4)
  700 IF(I .EQ. M) GO TO 800
      KK=JC+1
  710 C2=1.0-CD
      S1=SD
  720 C1=C2
      S2=S1
      KK=KK+KSPAN
  730 AK=A(KK)
      A(KK)=C2*AK-S2*B(KK)
      B(KK)=S2*AK+C2*B(KK)
      KK=KK+KSPNN
      IF(KK .LE. NT) GO TO 730
      AK=S1*S2
      S2=S1*C2+C1*S2
      C2=C1*C2-AK
      KK=KK-NT+KSPAN
      IF(KK .LE. KSPNN) GO TO 730
      C2=C1-(CD*C1+SD*S1)
      S1=S1+(SD*C1-CD*S1)
      C1=2.0-(C2**2+S1**2)
      S1=C1*S1
      C2=C1*C2
      KK=KK-KSPNN+JC
      IF(KK .LE. KSPAN) GO TO 720
      KK=KK-KSPAN+JC+INC
      IF(KK .LE. JC+JC) GO TO 710
      GO TO 100
C  PERMUTE THE RESULTS TO NORMAL ORDER---DONE IN TWO STAGES
C  PERMUTATION FOR SQUARE FACTORS OF N
  800 NP(1)=KS
      IF(KT .EQ. 0) GO TO 890
      K=KT+KT+1
      IF(M .LT. K) K=K-1
      J=1
      NP(K+1)=JC
  810 NP(J+1)=NP(J)/NFAC(J)
      NP(K)=NP(K+1)*NFAC(J)
      J=J+1
      K=K-1
      IF(J .LT. K) GO TO 810
      K3=NP(K+1)
      KSPAN=NP(2)
      KK=JC+1
      K2=KSPAN+1
      J=1
      IF(N .NE. NTOT) GO TO 850
C  PERMUTATION FOR SINGLE-VARIATE TRANSFORM (OPTIONAL CODE)
  820 AK=A(KK)
      A(KK)=A(K2)
      A(K2)=AK
      BK=B(KK)
      B(KK)=B(K2)
      B(K2)=BK
      KK=KK+INC
      K2=KSPAN+K2
      IF(K2 .LT. KS) GO TO 820
  830 K2=K2-NP(J)
      J=J+1
      K2=NP(J+1)+K2
      IF(K2 .GT. NP(J)) GO TO 830
      J=1
  840 IF(KK .LT. K2) GO TO 820
      KK=KK+INC
      K2=KSPAN+K2
      IF(K2 .LT. KS) GO TO 840
      IF(KK .LT. KS) GO TO 830
      JC=K3
      GO TO 890
C  PERMUTATION FOR MULTIVARIATE TRANSFORM
  850 K=KK+JC
  860 AK=A(KK)
      A(KK)=A(K2)
      A(K2)=AK
      BK=B(KK)
      B(KK)=B(K2)
      B(K2)=BK
      KK=KK+INC
      K2=K2+INC
      IF(KK .LT. K) GO TO 860
      KK=KK+KS-JC
      K2=K2+KS-JC
      IF(KK .LT. NT) GO TO 850
      K2=K2-NT+KSPAN
```

```
      KK=KK-NT+JC
      IF(K2 .LT. KS) GO TO 850
  870 K2=K2-NP(J)
      J=J+1
      K2=NP(J+1)+K2
      IF(K2 .GT. NP(J)) GO TO 870
      J=1
  880 IF(KK .LT. K2) GO TO 850
      KK=KK+JC
      K2=KSPAN+K2
      IF(K2 .LT. KS) GO TO 880
      IF(KK .LT. KS) GO TO 870
      JC=K3
  890 IF(2*KT+1 .GE. M) RETURN
      KSPNN=NP(KT+1)
C  PERMUTATION FOR SQUARE-FREE FACTORS OF N
      J=M-KT
      NFAC(J+1)=1
  900 NFAC(J)=NFAC(J)*NFAC(J+1)
      J=J-1
      IF(J .NE. KT) GO TO 900
      KT=KT+1
      NN=NFAC(KT)-1
      IF(NN .GT. MAXP) GO TO 998
      JJ=0
      J=0
      GO TO 906
  902 JJ=JJ-K2
      K2=KK
      K=K+1
      KK=NFAC(K)
  904 JJ=KK+JJ
      IF(JJ .GE. K2) GO TO 902
      NP(J)=JJ
  906 K2=NFAC(KT)
      K=KT+1
      KK=NFAC(K)
      J=J+1
      IF(J .LE. NN) GO TO 904
C  DETERMINE THE PERMUTATION CYCLES OF LENGTH GREATER THAN 1
      J=0
      GO TO 914
  910 K=KK
      KK=NP(K)
      NP(K)=-KK
      IF(KK .NE. J) GO TO 910
      K3=KK
  914 J=J+1
      KK=NP(J)
      IF(KK .LT. 0) GO TO 914
      IF(KK .NE. J) GO TO 910
      NP(J)=-J
      IF(J .NE. NN) GO TO 914
      MAXF=INC*MAXF
C  REORDER A AND B, FOLLOWING THE PERMUTATION CYCLES
      GO TO 950
  924 J=J-1
      IF(NP(J) .LT. 0) GO TO 924
      JJ=JC
  926 KSPAN=JJ
      IF(JJ .GT. MAXF) KSPAN=MAXF
      JJ=JJ-KSPAN
      K=NP(J)
      KK=JC*K+II+JJ
      K1=KK+KSPAN
      K2=0
  928 K2=K2+1
      AT(K2)=A(K1)
      BT(K2)=B(K1)
      K1=K1-INC
      IF(K1 .NE. KK) GO TO 928
  932 K1=KK+KSPAN
      K2=K1-JC*(K+NP(K))
      K=-NP(K)
  936 A(K1)=A(K2)
      B(K1)=B(K2)
      K1=K1-INC
      K2=K2-INC
      IF(K1 .NE. KK) GO TO 936
      KK=K2
      IF(K .NE. J) GO TO 932
      K1=KK+KSPAN
      K2=0
  940 K2=K2+1
      A(K1)=AT(K2)
      B(K1)=BT(K2)
      K1=K1-INC
      IF(K1 .NE. KK) GO TO 940
      IF(JJ .NE. 0) GO TO 926
      IF(J .NE. 1) GO TO 924
  950 J=K3+1
      NT=NT-KSPNN
      II=NT-INC+1
      IF(NT .GE. 0) GO TO 924
      RETURN
C  ERROR FINISH, INSUFFICIENT ARRAY STORAGE
  998 ISN=0
      PRINT 999
      STOP
  999 FORMAT(44H0ARRAY BOUNDS EXCEEDED WITHIN SUBROUTINE FFT)
      END
      FUNCTION F1(R,NP,W)
C*****************************************************
C
C    FUNCTION TO DETERMINE THE NUMERATOR FUNCTION IN NEWTAR
C    RADIAL FREQUENCY W.
C
C    INPUT :
C              NP,R(1),...,R(NP)
C              W
C
C*****************************************************
C
      DIMENSION R(NP)
C
      C=0.
      DO 1 I=1,NP
      DI=FLOAT(I)
    1 C=C+DI*R(I)*SIN(DI*W)
      F1=C
C
      RETURN
      END
      FUNCTION F1P(R,NP,W)
C*****************************************************
C
C    FUNCTION TO DETERMINE DERIVATIVE OF F1 AT RADIAL
C    FREQUENCY W
C
C    INPUT :
C              NP,R(1),...,R(NP)
C              W
```

```
C****************************************************
C
      DIMENSION R(NP)
C
      C=0.
      DO 1 I=1,NP
      DI=FLOAT(I)
    1 C=C+DI*DI*R(I)*COS(DI*W)
      FIP=C
C
      RETURN
      END
      SUBROUTINE FNFREQ(N,NCOVM,NFREQS)
C************************************************************
C
C    SUBROUTINE TO AUTOMATICALLY SPECIFY A VALUE FOR Q (=NFREQS)
C    THAT SATISFIES THE REQUIREMENTS FOR FFT AND THAT IS AT LEAST
C    AS LARGE AS N+NCOVM.
C
C************************************************************
      DIMENSION NFQ(11)
      DATA NFQ/60,96,144,192,288,384,480,576,768,960,1152/
      NST=N+NCOVM
      IF(NST.LE.1152) GO TO 10
      NFREQS=0
      RETURN
   10 I=1
   20 IF(NST.LE.NFQ(I)) GO TO 30
      I=I+1
      GO TO 20
   30 NFREQS=NFQ(I)
      RETURN
      END
      SUBROUTINE FORIER(F,U,N,A,MA)
C**************************************************************
C    SUBROUTINE TO COMPUTE THE FOURIER TRANSFORM
C    PHI(V) OF A DENSITY DEFINED ON (0,1) FOR V=0,1, . . ,M
C    INPUT :
C             F,U,N   VECTORS OF LENGTH N CONTAINING F(U),U
C             MA : MAXIMUM VALUE OF V FOR WHICH PHI(V) IS COMPUTED
C    OUTPUT   A : COMPLEX-VALUED VECTOR CONTAINING THE PHI'S
C    SUBROUTINES CALLED : NONE
C
C    NOTE: THIS ROUTINE IS AN INVERSE FOURIER TRANSFORM DESIGNED
C          TO PRODUCE THE AUTOCORRELATION FUNCTION OF A TIME
C          SERIES GIVEN THE SPECTRAL DENSITY FUNCTION.
C
C*****************************************************************
      DIMENSION F(N),U(N)
      COMPLEX A(MA),Z
      TWOPI=8.*ATAN(1.)
      FN=FLOAT(N)
      DO 20 IM=1,MA
      FIM=IM-1
      A(IM)=CMPLX(0.,0.)
      DO 10 I=1,N
         Z=CMPLX(0.,TWOPI*FIM*U(I))
   10    A(IM)=A(IM)+F(I)*CEXP(Z)
      A(IM)=A(IM)/FLOAT(N)
   20 CONTINUE
      RETURN
      END
      FUNCTION FOFNC(X,IDOH)
C*****************************************************
C  ROUTINE TO COMPUTE THE VARIOUS DENSITY-QUANTILE FUNCTIONS
C       INPUT:
C                X - VALUE AT WHICH THE FUNCTION IS TO BE COMPUTED
C                IDOH - INDICATOR FOR THE DESIRED FUNCTION,
C                       MUST BE IN THE EXCLUSIVE RANGE 1-11
C*****************************************************
      COMMON / CNSTNT / PI,TWOPI,RSQTPI,RPI
      DATA EPS / 1.E-14 /
      GO TO(1,2,3,4,5,6,7,8,9,10,11),IDOH
C  COMPUTE THE NORMAL
 1    CONTINUE
      IF( X .LT. .001) GO TO 101
      IF( X .GT. .999) GO TO 101
      CALL MDNRIS(X,T,IER)
      FOFNC=EXP(-0.5*T*T) * RSQTPI
      RETURN
  101 FOFNC = EPS
      RETURN
C  COMPUTE THE EXPONENTIAL
 2    CONTINUE
      FOFNC=1.-X
      RETURN
C  COMPUTE THE LOGISTIC
 3    CONTINUE
      FOFNC = X*(1. - X)
      RETURN
C  COMPUTE THE DOUBLE EXPONENTIAL
 4    CONTINUE
      FOFNC = .5 - ABS(X-.5)
      RETURN
C  COMPUTE THE UNIFORM RECIPROCAL
 5    CONTINUE
      FOFNC = (1. - X)**2
      RETURN
C  COMPUTE THE CAUCHY
 6    CONTINUE
      FOFNC = SIN(PI*X)**2 * RPI
      RETURN
C  COMPUTE THE EXTREME VALUE
 7    CONTINUE
      IF(X .EQ. 1.) GO TO 102
      FOFNC = (X - 1.)*ALOG(1. - X)
      RETURN
  102 CONTINUE
      FOFNC = EPS
      RETURN
C  COMPUTE THE LOG NORMAL
 8    CONTINUE
      IF(X .EQ. 1.) GO TO 103
      IF(X .EQ. 0.) GO TO 103
      CALL MDNRIS(X,T,IER)
      FOFNC = T * (.5*T+1) * RSQTPI
      FOFNC = EXP(-FOFNC)
      RETURN
  103 FOFNC = EPS
      RETURN
C  COMPUTE THE PARETO
 9    CONTINUE
      FOFNC = (1.-X)**(1.+BETAP)/BETAP
      RETURN
C  COMPUTE THE WEIBULL
 10   CONTINUE
      IF(X .EQ. 1.) GO TO 104
      FOFNC = (1.-X)*(-ALOG(1.-X))**(1.-BETAW)/BETAW
      RETURN
  104 CONTINUE
      FOFNC = EPS
      RETURN
```

```
C   COMPUTE THE HALF LOGISTIC
 11    CONTINUE
       FOFNC=1.-X**2
 99    RETURN
       END
       SUBROUTINE FTERP(U,V,X,F,N,M)
C*************************************************************
C
C     SUBROUTINE TO PERFORM LINEAR INTERPOLATION ON V TO OBTAIN F AT
C     THE M X VALUES.
C
C     INPUT: U - VECTOR OF VALUES AT WHICH V IS EVALUATED
C            V - FUNCTION VALUES TO INTERPOLATE
C            X - VALUES AT WHICH INTERPOLATED FFUNCTION TO BE
C                EVALUATED
C            N - DIMENSION OF VECTORS U AND V
C            M - DIMENSION OF VECTORS X AND F
C
C     NOTE. ALL ABSCISSA VECTORS MUST BE ORDERED
C
C     OUTPUT: F - INTERPOLATED FUNCTION VALUES
C
C*************************************************************
C
       DIMENSION U(N),V(N),X(M),F(M)
       IF(N.EQ.M) GO TO 100
       II=1
       DO 60 I=1,M
   10  IF(X(I)-U(II))20,40,50
   20  IF(II.NE.1) GO TO 30
       F(I)=V(1)+(V(2)-V(1))*(X(I)-U(1))/(U(2)-U(1))
       GO TO 60
   30  F(I)=V(II-1)+(V(II)-V(II-1))*(X(I)-U(II-1))/(U(II)-U(II-1))
       GO TO 60
   40  F(I)=V(II)
       GO TO 60
   50  II=II+1
       IF(II.LT.N) GO TO 10
       II=N
       GO TO 30
   60  CONTINUE
  100  RETURN
       END
       SUBROUTINE ICODEA(K,IFORM,NAME)
C*************************************************
C     SUBROUTINE TO CONVERT INTEGER VARIABLE K
C     WHICH HAS 8 CHARACTER I-FORMAT IFORM
C     TO 8 CHARACTER ALPHAMERIC ARRAY NAME WHICH IS
C     IN A-FORMAT.
C     INPUT  : NSCRCH : SCRATCH TAPE NUMBER
C              K, IFORM
C     OUTPUT : NAME(1),NAME(2) : 4 CHARACTERS EACH
C*************************************************
       COMMON /UNIT/ IUNIT,NSCRCH
       DIMENSION NAME(2),IFORM(2)
       REWIND NSCRCH
       WRITE(NSCRCH,IFORM)K
       REWIND NSCRCH
       READ(NSCRCH,10)NAME
    10 FORMAT(2A4)
       RETURN
       END
       SUBROUTINE KSD(D,U,N,DM,UM,DP,UP)
C*************************************************
C     SUBROUTINE TO COMPUTE KOMOLGOROV-SMIRNOFF STATISTIC FOR
C     THE DEVIATIONS D(U)-U.  UPPER AND LOWER BOUNDS ARE GIVEN.
C     INPUT : D,U,N
C     OUTPUT :
C             DP,UP : MAX (+) DEVIATION, DP, WHICH IS AT U=UP
C             DM,UM : MAX (-) DEVIATION, DM, WHICH IS AT U=UM
C     SUBROUTINES CALLED   NONE
C*************************************************
       DIMENSION D(N),U(N)
       SON = SQRT(FLOAT(N))
       DP = (D(1) - U(1))
       UP = U(1)
       DM = DP
       UM = UP
       DO 10   I = 2,N
       DT = (D(I) - U(I) )
       IF(DT .LE. DP) GOTO 1
       UP = U(I)
       DP = DT
       GOTO 10
 1     IF(DT .GE. DM) GOTO 10
       UM = U(I)
       DM = DT
    10 CONTINUE
       DP = SON*DP
       DM = SON*DM
       RETURN
       END
       SUBROUTINE MACV(BETA,SIG,NO,R,RO)
C*************************************************************
C
C     SUBROUTINE TO CALCULATE THE AUTOCOVARIANCES RO,R(1),
C        .R(NO) FOR A MOVING AVERAGE PROCESS OF ORDER NO
C     WITH PARAMETERS BETA(1),....,BETA(NO), AND SIG (RES VAR)
C
C     INPUT :
C               NO,BETA(1),....,BETA(NO),SIG
C
C     OUTPUT :
C               RO,R(1),....,R(NO)
C
C     SUBROUTINES CALLED : NONE
C
C*************************************************************
C
       DIMENSION BETA(NO),R(NO)
C
       C=1
       DO 1 I=1,NO
    1  C=C+BETA(I)*BETA(I)
       RO=C*SIG
C
       DO 2 I=1,NO
          C=BETA(I)
          IF(I.EQ.NO) GO TO 2
          NOMI=NO-I
          DO 3 J=1,NOMI
    3     C=C+BETA(J)*BETA(J+I)
    2  R(I)=C*SIG
C
       RETURN
       END
       SUBROUTINE MAMXSL(RY,RYO,BETA,MR,NDIM,K1,K2,KTOP,
      1L,A,KP,NAME,COF,IND,RVAR,IER)
C*************************************************************
C
C     SUBROUTINE TO PERFORM SUBSET MIXED SCHEME ESTIMATION
C           (G(L)Y(T)=H(L)E(T))
```

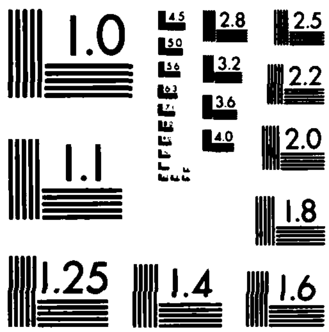
```
C
C     INPUT :
C             MR,RVO,RY(1),....,RY(MR) : 1ST MR+1 AUTOCOVARIANCES
C             OF Y
C        NOTE : IF MR IS NEGATIVE, THE COVARIANCES OBTAINED
C                ARE NOT PRINTED
C             NDIM : DIMENSION OF MATRIX A IN CALLING
C             PROGRAM
C             BETA(1),....BETA(MR) : COEFFICIENTS OF MRTH
C             ORDER MOVING AVERAGE REPRESENTATION OF Y
C             RVAR: STANDARDIZED RESIDUAL VARIANCE FOR MA MODEL
C
C     K1 K2 :
C
C             NUMBER OF LAGS OF Y AND E RESPECTIVELY TO BE
C             INCLUDED IN THE FULL MODEL.
C             IF(K1.LT.MR-MA), THE MOST IMPORTANT LAGS OF Y
C             ARE CHOSEN FOR INCLUSION BY THE MAGNITUDE OF
C             THEIR AUTOCOVARIANCE. THE SAME PROCESS IS
C             CARRIED OUT FOR LAGS OF E IF(K2.LT.MR-MA).
C             IF(K1.EQ.0) THE RESULT IS A PURE MOVING
C             AVERAGE STEPWISE REGRESSION. IF(K2.EQ.0), THE
C             RESULT IS A PURE AUTOREGRESSIVE STEPWISE
C             REGRESSION.
C             IF(K1.EQ.-J1), OR (K2.EQ.-J2) OR BOTH, THE LAGS
C             SPECIFIED BY USING THE VECTOR NAME AS INPUT
C             (LAGS OF Y BEING POSITIVE AND IN THE FIRST
C             J1 POSITIONS. THE LAGS OF E BEING NEGATIVE
C             AND IN THE NEXT J2 POSITIONS) ARE FORCED INTO THE MODEL.
C
C             KTOP : INDICATOR (IF KTOP = 0, THE BEST MODEL
C             IS CHOSEN AUTOMATICALLY, IF KTOP = J, THE J
C             MOST IMPORTANT LAGS ARE FORCED INTO THE MODEL)
C             L : SAMPLE SIZE
C
C     OUTPUT :
C             KP   NUMBER OF PREDICTORS IN CHOSEN MODEL
C             NAME(1),... ,NAME(K1+K2) . VECTOR OF LAGS IN FULL
C             MODEL
C
C     NOTE : IN VECTOR NAME, YLAGS ARE POSITIVE, E LAGS ARE
C     NEGATIVE
C
C             COF : COEFFICIENTS OF CHOSEN LAGS
C             IND : LAGS CHOSEN (ACCORDING TO NAME)
C             IER (1 MEANS ALL DIAGONALS ARE LT 1.E-5 FOR
C             SOME CYCLE. 0 IS NORMAL RETURN)
C
C     SUBROUTINES CALLED : SEIREG,MAX
C
C*************************************************************
C
      DIMENSION RY(1),BETA(MR),A(NDIM,NDIM),NAME(1),COF(1),
     1IND(1),REY(300),REE(300),WK(300),NWK(300)
      DIMENSION RYE(300)
      DATA NOUT/6/
C
C     FIND AUTOCOVARIANCES :
C
      MOPT=MR
      IOPT=0
      IF(MR.GT.0) GO TO 110
      IOPT=1
      MR=-MR
  110 CONTINUE
      IF(MR.GT.1) GO TO 103
      WRITE(NOUT,104)
  104 FORMAT(10X,'MR.LT.1 IN CVMXSL')
      GO TO 99
  103 CONTINUE
C
      DO 105 I=1,MR
      REY(I)=RVAR*BETA(I)
      RYE(I)=0.0
      REE(I)=0.0
  105 CONTINUE
      REYO=RVAR
      REEO=RVAR
      IF(IOPT.EQ.1) GO TO 109
      WRITE(NOUT,106)
  106 FORMAT(1H1,9X,2X,3HLAG,10X,6HRYY(V),10X,6HREY(V),
     110X,6HRYE(V),10X,6HREE(V),10X,
     16HPVH(V)/10X,89(1H-))
      II=0
      WRITE(NOUT,107) II,RYO,REYO,REYO,REEO,REYO
  107 FORMAT(10X,I5,2X,5(F14.7,2X))
      PVH=REYO
      DO 108 I=1,MR
      PVH=PVH+(REY(I)**2)/REYO
  108 IF(PVH.LE.0.9999) WRITE(NOUT,107) I,RY(I),REY(I),RYE(I)
     +REE(I)),PVH
C
C
C     DETERMINE VECTOR NAME :
C
C
  109 KK=IABS(K1)+IABS(K2)
      IF(KK.EQ.0) RETURN
      DO 1 I=1,KK
      DO 1 J=1,KK
    1 A(I,J)=0
      M1=MR
C
C     K1 :
C
      IF(K1.GT.0) GO TO 10
C
C     K1.LE.0 :
C
      K1=IABS(K1)
      GO TO 20
   10 IF(K1.EQ.M1) GO TO 15
C
C     0.LT.K1.LT.M1 :
C
      DO 11 I=1,M1
      WK(I)=ABS(RY(I))
   11 NWK(I)=0
      DO 12 I=1,K1
      CALL MAX(WK,M1,WKMAX,MIND)
      NWK(MIND)=1
   12 WK(MIND)=0.
      II=1
      DO 13 I=1,M1
      IF(NWK(I).EQ.0) GO TO 13
      NAME(K1-II+1)=I
      II=II+1
   13 CONTINUE
      GO TO 20
C
C     K1.EQ.M1 :
C
```

```
   15   DO 16 I=1,K1
   16   NAME(I)=K1-I+1
C
C    K2 :
C
   20   IF(K2.GT.0) GO TO 30
C
C    K2.LE.0 .
C
        K2=IABS(K2)
        GO TO 40
   30   IF(K2.EQ.M1) GO TO 35
C
C    0.LT.K2.LT.M1 :
C
        DO 31 I=1,M1
        WK(I)=ABS(REV(I))
   31   NWK(I)=0
        DO 32 I=1,K2
        CALL MAX(WK,M1,WKMAX,MIND)
        NWK(MIND)=1
   32   WK(MIND)=0.
        I1=1
        DO 33 I=1,M1
        IF(NWK(I).EQ.0) GO TO 33
        NAME(K1+K2-I1+1)=-I
        I1=I1+1
   33   CONTINUE
        GO TO 40
C
C    K2.EQ.M1 .
C
   35   DO 36 I=1,K2
   36   NAME(K1+I)=I-K2-1
C
C
C    FORM MATRIX A :
C
C
   40   IF((K1.NE.0).AND.(K2.NE.0)) GO TO 50
        IF(K2.EQ.0) GO TO 45
C
C    PURE MOVING AVERAGE :
C
        DO 41 I=1,K2
        I1=NAME(I)
        DO 41 J=1,I
        I2=IABS(I1-NAME(J))
        IF(I2.EQ.0) GO TO 42
        A(I,J)=REE(I2)
        GO TO 41
   42   A(I,J)=REEO
   41   A(J,I)=A(I,J)
        GO TO 60
C
C    PURE AUTOREGRESSION :
C
   45   DO 46 I=1,K1
        I1=NAME(I)
        DO 46 J=1,I
        I2=IABS(I1-NAME(J))
        IF(I2.EQ.0) GO TO 47
        A(I,J)=RY(I2)
        GO TO 46
   47   A(I,J)=RYO
   46   A(J,I)=A(I,J)
        GO TO 60
C
C    MIXED .
C
   50   DO 54 I=1,KK
        I1=NAME(I)
        DO 54 J=1,I
        J1=NAME(J)
        I2=I1*J1
        I3=IABS(I1)
        I4=IABS(J1)
        I3=IABS(I3-I4)
        IF(I2.GT.0) GO TO 56
C    REV :
        IF(I3.EQ.0) GO TO 55
        A(I,J)=REV(I3)
        IF((I1+J1).GT.0) A(I,J)=RVE(I3)
        GO TO 54
   55   A(I,J)=REVO
        GO TO 54
   56   IF(I1.GT.0) GO TO 58
C    REE
        IF(I3.EQ.0) GO TO 57
        A(I,J)=REE(I3)
        GO TO 54
   57   A(I,J)=REEO
        GO TO 54
C    RYY .
   58   IF(I3.EQ.0) GO TO 59
        A(I,J)=RY(I3)
        GO TO 54
   59   A(I,J)=RYO
C
C
   54   A(J,I)=A(I,J)
C
   60   N=KK+1
        A(N,N)=RYO
        DO 61 I=1,KK
        I1=NAME(I)
        IF(I1.GT.0) GO TO 63
        I2=-I1
        A(I,N)=REV(I2)
        GO TO 61
   63   A(I,N)=RY(I1)
   61   A(N,I)=A(I,N)
C
C    CALL SELREG :
C
C    INPUT TWICE SAMPLE SIZE FOR ARGUMENT 6 TO PERMIT MORE
C    VARIABLES TO BE SIGNIFICANT.
C
        L2=L*2
        CALL SELREG(A,NDIM,KK,NAME,KTOP,L2,KP,COF,IND,IER)
        IF(KP.EQ.0) GO TO 99
        DO 70 I=1,KP
        IF(IND(I).GT.0) COF(I)=-COF(I)
   70   CONTINUE
   99   CONTINUE
        MR=MOPT
C
C
C
        RETURN
        END
        SUBROUTINE MAX(X,N,XMAX,IND)
```

```fortran
C*****************************************************
C
C     SUBROUTINE TO FIND THE MAXIMUM VALUE (XMAX) AND THE
C     INDEX OF THE MAXIMUM VALUE (IND) OF A VECTOR X OF
C     LENGTH N.
C
C     INPUT :
C              N,X(1),....,X(N)
C
C     OUTPUT :
C              XMAX,IND
C
C     SUBROUTINES CALLED : NONE
C
C*****************************************************
C
      DIMENSION X(N)
C
      XMAX=X(1)
      IND=1
      IF(N.EQ.1) RETURN
C
      DO 1 I=2,N
      IF(XMAX.LT.X(I)) IND=I
    1 XMAX=X(IND)
C
      RETURN
      END
      SUBROUTINE MAXSPA(SPEC,ALPH,NFREQS,NP,MAXIT,DEL,NOFM,P,IER)
C*****************************************************
C
C     SUBROUTINE TO OBTAIN THE NOFM PERIODS, P(1),....P(NOFM)
C     OF THE MAXIMA IN (0,.25) OF THE AUTOREGRESSIVE SPECTRA
C     SPEC(1),....,SPEC(NFREQS) CORRESPOMDING TO THE AUTO-
C     REGRESSIVE COEFFICIENTS ALPH(1),....,ALPH(NP).
C
C     INPUT :
C              NP,ALPH(1),....,ALPH(NP)
C              NFREQS : NUMBER OF EQUALLY SPACED FREQUENCIES
C              BETWEEN 0 AND TWOPI AT WHICH SPEC IS EVALUATED
C              SPEC(1),....,SPEC(NFREQS)
C              MAXIT : MAXIMUM NUMBER OF ITERATIONS IN NEWTAR
C                   IF MAXIT IS NEGATIVE, NO PRINTING IS DONE
C              DEL : CONVERGENCE CRITERION IN NEWTAR
C        NOTE : THE RES VAR FOR THE AR PROCESS IS INPUTTED IN
C        SPEC(NFREQS)
C
C     OUTPUT :
C              NOFM : NUMBER OF MAXIMA FOUND (NOFM.LE.NP)
C              P(1),....,P(NOFM) : PERIODS OF MAXIMA
C              IER(1),....,IER(NOFM) : RETURN CODE FOR EACH MAXIMA
C              (SEE NEWTAR WRITEUP)
C
C     SUBPROGRAMS CALLED : NEWTAR,MACV,FI,FIP
C
C*****************************************************
C
      DIMENSION SPEC(NFREQS),ALPH(NP),ST(250),D(250),
     1R(250),P(NP),IER(NP)
      DATA NOUT/6/
C
C
      TWOPI=8.*ATAN(1.0)
      DN=FLOAT(NFREQS)
      NOFM=0
      N1=NFREQS/4
      IOPT=0
      MOPT=MAXIT
      IF(MAXIT.GT.0) GO TO 110
      MAXIT=-MAXIT
      IOPT=1
  110 CONTINUE
C
C    FIND STARTING VALUES :
C
      DO 1 I=2,N1
      IF((SPEC(I-1).LT.SPEC(I)).AND.(SPEC(I).GT.SPEC(I+1))) GO TO 2
      GO TO 1
    2 IF(NOFM.EQ.NP) GO TO 99
      NOFM=NOFM+1
      ST(NOFM)=(FLOAT(I-1)*TWOPI)/DN
      D(NOFM)=SPEC(I)
    1 CONTINUE
C
C
      IF(NOFM.EQ.0) GO TO 100
C
C
   99 IF(IOPT.EQ.1) GO TO 115
      WRITE(NOUT,5)
    5 FORMAT(//,10X,'ANALYSIS OF AR SPECTRAL MAXIMA :')
      WRITE(NOUT,6)
    6 FORMAT(14X,1HI,10X,5HSTART,10X,5HFINAL,5X,5HNUMIT,
     18X,7HSPEC(I),2X,3HIER/10X,65(1H-))
  115 CONTINUE
C
C    FIND AUTOCOVARIANCES :
C
      CALL MACV(ALPH,1.,NP,R,R0)
C
C    FIND MAXIMA :
C
      DO 10 I=1,NOFM
      CALL NEWTAR(ST(I),R,NP,MAXIT,DEL,P(I),NUMIT,IER(I))
      IF(IER(I).NE.1) GO TO 15
      PP=P(I)
      C=R0
      DO 30 J=1,NP
   30 C=C+2.*R(J)*COS(FLOAT(J)*PP)
      D(I)=SPEC(NFREQS)/(TWOPI*C)
      ST(I)=TWOPI/ST(I)
      P(I)=TWOPI/P(I)
   15 IF(IOPT.EQ.1) GO TO 10
      WRITE(NOUT,16) I,ST(I),P(I),NUMIT,D(I),IER(I)
   16 FORMAT(10X,I5,2X,F13.7,2X,F13.7,I10,2X,F13.7,I6)
   10 CONTINUE
      DO 50 I=1,NOFM
      CALL MAX(D,NOFM,DMAX,MMIN)
      ST(I)=P(MMIN)
   50 D(MMIN)=0.
      DO 51 I=1,NOFM
   51 P(I)=ST(I)
C
C
  100 CONTINUE
      MAXIT=MOPT
C
C    RETURN FOR NO MAXIMA :
C
      RETURN
      END
      SUBROUTINE MONRIS (P,Y,IER)
```

AD-A132 217 A USER'S GUIDE TO ARSPIQ (AUTOREGRESSIVE SPECTRAL INFORMATION QUANTILE IDENTIFICATION)(U) TEXAS A AND M UNIV COLLEGE STATION DEPT OF STATISTICS T J WOODFIELD AUG 83 A-25 DAAG29-83-K-0051 2/2

UNCLASSIFIED F/G 9/2 NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```
C***************************************************************
C    QUANTILE FUNCTION FOR N(0,1)
C***************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      REAL*4 A(65),C(17),D(26),E(23),H1,H2,H3,H4,R2
      EQUIVALENCE (A(24),C(1)),(A(41),D(1)),  (A(1),E(1) )
      DATA E/.992885376618940,.120467616143104,
     *      .016078189342100,.002686704437182,
     *      .000499634730236,.000098896218599,
     *      .000020391812764,.000004327271618,
     *      .000000938081413,.000000206734721,
     *      .000000046159699,.000000010416680,
     *      .000000002371801,.000000000543928,
     *      .000000000125549,.000000000029138,
     *      .000000000006795,.000000000001591,
     *      .000000000000374,.000000000000088,
     *      .000000000000021,.000000000000005,
     *      .000000000000001/
      DATA C/.912158803417554,-.016266281867664,
     *      .000433556472949,.000214438570074,
     *      .000002625751076,-.000003021091050,
     *      -.000000012406062,.000000062406609,
     *      -.000000000540125,-.000000001423208,
     *      .000000000034384,.000000000033585,
     *      -.000000000001458,-.000000000000810,
     *      .000000000000053,.000000000000020,
     *      -.000000000000002/
      DATA D/.956679709020493,-.023107004309065,
     *      -.004374236097508,-.000576503422651,
     *      -.000010961022307,.000025108547025,
     *      .000010562336068,.000002754412330,
     *      .000000432484498,-.000000020530337,
     *      -.000000043891537,-.000000017684010,
     *      -.000000003991289,-.000000000186932,
     *      .000000000272923,.000000000132817,
     *      .000000000031834,.000000000001670,
     *      -.000000000002036,-.000000000000965,
     *      -.000000000000220,-.000000000000010,
     *      .000000000000013,.000000000000006,
     *      .000000000000001/
      DATA H1,H2,H3,H4,R2/-1.5488130423733,
     *      2.5654901231478,-.55945763132983,
     *      2.2879157162634,-1.4142135623731/
      DATA XINF / 1.E10 /
      X = 1.-P-P
      IER = 0
      SIGMA = SIGN(1.,X)
      IF( .NOT.(X.GT.-1..AND.X.LT.1.)) GO TO 45
      Z = ABS(X)
      IF(Z .GT. .8) GO TO 25
      W = Z*Z/.32-1.
      N = 22
      IPP = 1
      L = 1
   10 LB2 = 1
      X3 = 1.
      X4 = W
      X6 = A(IPP)
   15 X6 = X6 + A(IPP+LB2) * X4
      X5 = X4 * W * 2.-X3
      X3 = X4
      X4 = X5
      LB2 = LB2 + 1
      IF(LB2 .LE. N) GO TO 15
      GO TO (20,35),L
   20 Y = Z * X6 * SIGMA
      GO TO 40
   25 B = SQRT(-ALOG(1.-Z*Z))
      IF(X .GT. .9975) GO TO 30
      W = H1*B+H2
      IPP = 24
      L = 2
      N = 16
      GO TO 10
   30 W = H3 * B + H4
      IPP = 41
      N = 24
      L = 2
      GO TO 10
   35 Y = B * X6 * SIGMA
   40 Y = R2*Y
      RETURN
   45 Y = SIGMA*XINF
      IER = 129
      WRITE(IUNIT,101)
  101 FORMAT(//////////1X,35H*****ERROR CONDITION IN MDNRIS*****  ///)
      RETURN
      END
      SUBROUTINE MIN(X,N,XMIN,IND)
C**************************************************************
C
C    SUBROUTINE TO FIND THE MINIMUM VALUE (XMIN) AND THE
C    INDEX OF THE MINIMUM VALUE (IND) OF A VECTOR X OF
C    LENGTH N.
C
C    INPUT :
C            N,X(1),....,X(N)
C
C    OUTPUT :
C            XMIN,IND
C
C    SUBROUTINES CALLED   NONE
C
C**************************************************************
C
      DIMENSION X(N)
C
      XMIN=X(1)
      IND=1
      IF(N.EQ.1) RETURN
C
      DO 1 I=2,N
      IF(XMIN.GT.X(I)) IND=I
    1 XMIN=X(IND)
C
      RETURN
      END
      SUBROUTINE MXSPSL(COF,IND,SIG,KP,NFREQS,CT,ST,SPEC)
C***************************************************************
C
C    SUBROUTINE TO CALCULATE SPECTRA FOR SELECT MIXED SCHEMES.
C
C    INPUT
C            KP : NUMBER OF VARIABLES IN MODEL
C            IND(1),....,IND(KP)   INDICES OF CHOSEN VARIABLES
C            (Y LAGS ARE POSITIVE, E LAGS NEGATIVE)
C            COF(1),....,COF(KP)   COEFFICIENTS OF CHOSEN
C            VARIABLES
C            SIG : RESIDUAL VARIANCE OF CHOSEN MODEL
C            NFREQS   NUMBER OF FREQUENCIES BETWEEN
C            0 AND TWOPI AT WHICH SPECTRA IS TO BE CALCULATED
C
```

```
C     OUTPUT :
C              SPEC(1),....,SPEC(NFREQS)
C
C     SUBROUTINES CALLED : FFT
C
C**************************************************
C
      DIMENSION COF(KP),IND(KP),CT(NFREQS),ST(NFREQS),
     1SPEC(NFREQS)
C
C
      FAC=SIG/(8.*ATAN(1.0))
      IF(KP.GT.0) GO TO 20
      DO 10 I=1,NFREQS
   10 SPEC(I)=FAC
      GO TO 99
C
C
   20 DO 30 I=1,NFREQS
      CT(I)=0
      ST(I)=0
   30 SPEC(I)=0
C
      CT(1)=1.
      ST(1)=1.
      DO 50 I=1,KP
      II=IND(I)
      IF(II.LT.0) GO TO 40
      ST(II+1)=COF(I)
      GO TO 50
   40 II=-II
      CT(II+1)=COF(I)
   50 CONTINUE
C
C
      CALL FFT(CT,SPEC,NFREQS,NFREQS,NFREQS,1)
C
      DO 60 I=1,NFREQS
      SPEC(I)=CT(I)*CT(I)+SPEC(I)*SPEC(I)
   60 CT(I)=0.
C
C
      CALL FFT(ST,CT,NFREQS,NFREQS,NFREQS,1)
C
      DO 70 I=1,NFREQS
   70 SPEC(I)=(FAC*SPEC(I))/(ST(I)*ST(I)+CT(I)*CT(I))
C
C
   99 CONTINUE
      RETURN
      END
      SUBROUTINE NEWTAR(ST,R,NP,MAXIT,DEL,Z,NUMIT,IER)
C**************************************************
C
C     SUBROUTINE TO DETERMINE THE RADIAL FREQUENCY Z
C     OF THE AUTOREGRESSIVE SPECTRA CORRESPONDING TO
C     R(1),....,R(NP) (R(.) OBTAINED BY CONVOLUTION OF
C     ALPH(1),....,ALPH(NP)) FOR STARTING RADIAL FREQUENCY
C     VALUE ST.
C
C     INPUT :
C              NP,ST,R(1),....,R(NP)
C              MAXIT : MAXIMUM NUMBER OF ITERATIONS IN
C              NEWTONS ALGORITHM
C              DEL : CONVERGENCE CRITERION
C
C     OUTPUT :
C              Z
C              NUMIT : NUMBER OF ITERATIONS NEEDED (IF
C              NONCONVEGENCE,NUMIT=MAXIT+1)
C                     IER : 1  NORMAL RETURN
C                           2  ZERO FREQUENCY ENCOUNTERED
C                           3  ZERO DENOMINATOR ENCOUNTERED
C                           4  NONCONVERGENCE INDICATOR
C
C     SUBPROGRAMS CALLED : FI,FIP
C
C**************************************************
C
      DIMENSION R(NP)
      DATA NOUT/6/
C
C     INITIALIZE :
C
      IER=1
      IT=1
      WO=ST
C
C     BEGIN ITERATION :
C
    1 IF(WO.LT.1.E-15) GO TO 4
      G=FI(R,NP,WO)
      GP=FIP(R,NP,WO)
      IF(ABS(GP).LT.1.E-25) GO TO 5
      WN=WO-(G/GP)
      C=ABS(WN-WO)/ABS(WO)
      IF(C.LT.DEL) GO TO 2
      IF(IT.EQ.MAXIT) GO TO 3
      IT=IT+1
      WO=WN
      GO TO 1
C
C     CONVERGENCE RETURN
C
    2 NUMIT=IT
      Z=WN
      RETURN
C
C     NONCONVERGENCE RETURN :
C
    3 NUMIT=MAXIT+1
      IER=4
      Z=WN
      RETURN
C
C     ERROR RETURN :
C
    4 WRITE(NOUT,10)
   10 FORMAT(10X,'ZERO FREQUENCY IN NEWTAR')
      IER=2
      RETURN
C
    5 WRITE(NOUT,15)
   15 FORMAT(10X,'ZERO DENOMINATOR IN NEWTAR')
      RETURN
C
      END
      SUBROUTINE PARTAR(PART,NP,ALPHA)
C*****************************************************
C
C     SUBROUTINE TO DETERMINE  AUTOREGRESSIVE COEFFICIENTS
C     ALPHA(1),....,ALPHA(NP) GIVEN PARTIAL AUTOCORRELATIONS
```

```
C     PART(1),...,PART(NP)
C
C     INPUT :
C           NP,PART(1),...,PART(NP)
C
C     OUTPUT :
C           ALPHA(1),...,ALPHA(NP)
C
C     SUBROUTINES CALLED : NONE
C
C*********************************************************
C
      DIMENSION PART(NP),ALPHA(NP)
C
C     NP=1 :
C
      ALPHA(1)=PART(1)
      IF(NP.EQ.1)GO TO 99
C
C     NP GT 1 :
C
C
C     FIX ORDER INP IN RECURSION :
C
      DO 20 INP=2,NP
C
C     FIND ALPHA(1),...,ALPHA(INP-1) :
C
      ALPHA(INP)=PART(INP)
      C=ALPHA(INP)
         NL=INP/2
         DO 10 J=1,NL
         INPMJ=INP-J
         TEMP=ALPHA(J)+C*ALPHA(INPMJ)
         ALPHA(INPMJ)=ALPHA(INPMJ)+C*ALPHA(J)
   10    ALPHA(J)=TEMP
   20 CONTINUE
C
C
   99 CONTINUE
      RETURN
      END
      SUBROUTINE PARTRV(PART,M,RO,RVAR)
C******************************************************
C
C     SUBROUTINE TO FIND BIASED RESIDUAL VARIANCES RVAR(1),....
C     RVAR(M), GIVEN PARTIAL AUTOCORRELATIONS PART(1),...,PART(M)
C     AND VARIANCE RO.
C
C     INPUT :
C             M,PART(1),...,PART(M),RO
C
C     OUTPUT :
C             RVAR(1),...,RVAR(M)
C
C     SUBROUTINES CALLED : NONE
C
C******************************************************
C
      DIMENSION PART(M),RVAR(M)
C
      RVAR(1)=RO*(1.-PART(1)**2)
      IF(M.EQ.1) RETURN
      DO 1 I=2,M
    1 RVAR(I)=RVAR(I-1)*(1.-PART(I)**2)
C
      RETURN
      END
      SUBROUTINE PARZ(RVAR,M,N,CAT,NORD)
C*****************************************************
C
C     SUBROUTINE TO DETERMINE THE ORDER OF AN AUTOREGRESSIVE
C     PROCESS BY PARZENS CAT CRITERIA
C
C     INPUT :
C             M,RVAR(1),...,RVAR(M) : STANDARDIZED RES VAR
C             FOR ORDERS 1 THRU M.
C             N : SAMPLE SIZE
C
C     OUTPUT :
C             NORD : DETERMINED ORDER
C             CAT(1),...,CAT(M)
C
C     SUBROUTINES CALLED : MIN
C
C*****************************************************
C
      DIMENSION RVAR(M),CAT(M)
C
      ON=FLOAT(N)
      DO 1 I=1,M
         C=0.
         DO 2 J=1,I
    2    C=C+((1.-(FLOAT(J)/ON))/RVAR(J))
         C=C/ON
    1 CAT(I)=C-((1.-(FLOAT(I)/ON))/RVAR(I))
      CALL MIN(CAT,M,CATMI,NORD)
      IF(CATMI.GT.-1.) NORD=0
C
      RETURN
      END
      SUBROUTINE PCORR(R,RO,M,CORR)
C******************************************************
C
C     SUBROUTINE TO TRANSFORM AUTOCOVARIANCES RO,R(1),...,R(M)
C     TO AUTOCORRELATIONS CORR(1),...,CORR(M) AND PLOT THE
C     VECTOR (1.,CORR(1),...,CORR(M-1) VIA CLPLT1.
C
C     INPUT :
C             M,RO,R(1),...,R(M)
C
C     OUTPUT :
C             CORR(1),...,CORR(M)
C
C     SUBROUTINES CALLED : CLPLT1,MAX,MIN,
C
C******************************************************
C
      DIMENSION R(M),CORR(M)
      DATA NOUT/6/
C
      WRITE(NOUT,2) RO
    2 FORMAT(10X,7HR(0) = ,E15.7)
      IF(M.GT.1) GO TO 10
      CORR(1)=R(1)/RO
      WRITE(NOUT,5) CORR(1)
    5 FORMAT(10X,'CORR(1) = ',E15.7)
      RETURN
   10 CONTINUE
      DO 1 I=1,M
    1 CORR(I)=R(I)/RO
      CALL CLPLTO(CORR,M,1,4HCORR,1.0,-1.0)
```

```
      RETURN
      END
      SUBROUTINE PLINE(NR,NC,NL,STRT,END,CHARL,PARRAY,VMIN,VMAX,VINC)
      DIMENSION STRT(NL),END(NL),CHARL(NL),PARRAY(NR,NC)
      NCM1 = NC - 1
      FNC = 1. / FLOAT(NCM1)
      NRP1 = NR + 1
C
      DO 10 IL = 1,NL
      IF ( STRT(IL) - END(IL) ) 20,30,40
 20   STRT(IL) = AMAX1(STRT(IL),VMIN)
      END(IL)  = AMIN1(END(IL) ,VMAX)
      GOTO 50
 40   STRT(IL) = AMIN1(STRT(IL),VMAX)
      END(IL)  = AMAX1(END(IL) ,VMIN)
 50   DS = (END(IL) - STRT(IL)) * FNC * VINC
      S  = (STRT(IL) - VMIN    ) * VINC + 1.5
      IND = INT(S)
      DO 60 I = 2,NCM1
      S = S + DS
      IF ( INT(S) .EQ. IND ) GOTO 60
      IND = INT(S)
      PARRAY(NRP1-IND,I) = CHARL(IL)
 60   CONTINUE
      GOTO 10
 30   IF ( STRT(IL) .LE. VMIN .OR. STRT(IL) .GE. VMAX) GOTO 10
      IND = (STRT(IL) - VMIN) * VINC + 1.5
      DO 70 I = 2,NCM1
      PARRAY(NRP1-IND,I) = CHARL(IL)
 70   CONTINUE
 10   CONTINUE
      RETURN
      END
      SUBROUTINE PPLOT(X,Y,NY,XCHAR,XZ,Z,NZ,IZ,ZCHAR,IORD,VMN,VMX,
     &                 NLINE,STRT,END,CHARL,LX,LY,LC,LI)
C*****************************************************************
C     ROUTINE TO DISPLAY PRINTER QUANTILE-BOX PLOT
C     INPUT :
C           X - VECTOR CONTAINING THE VALUES J/(NY+1) WHERE
C               J=1,2,3,....,NY
C           Y - VECTOR OF SIZE NY TO BE PLOTTED
C           XCHAR - CHARACTER FOR Y IN PLOT
C           LX,LY - VECTORS OF SIZE 2 CONTAINING THE LABELS
C                   FOR X AND Y RESPECTIVELY
C           LC - VECTOR OF SIZE 20 CONTAINING THE CAPTION FOR
C                THE PLOT
C           Z - OPTIONAL VECTOR OF SIZE NZ TO BE PLOTTED
C           XZ - ABSCISSA FOR Z
C           NLINE,STRT,END - VALUES FOR SUB. PLINE WHEN Z IS A LINE
C           VMN,VMX - MIN AND MAX VALUES FOR ORDINATE OF PLOT
C           IBOX - EQUAL ZERO IF BOX PLOTS ARE NOT WANTED
C           IZ - EQUAL 1 IF VECTOR Z IS TO BE PLOTTED
C           ZCHAR - PLOTTING CHARACTER TO BE USED FOR THE
C                   Z VECTOR. MUST BE DIFFERENT FROM O.
C           LI - OPTIONAL LABEL TO FOLLOW CAPTION
C           IORD = 1  IF Y IS ORDERED FROM MIN TO MAX
C                = 0  IF Y IS NOT ORDERED
C                = 2 FOR HORIZONTAL ZERO-LINE (FOR IQ PLOT)
C*****************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION PARRAY(51,61),CHAR(10),X(NY),Y(NY),LX(2),LY(2),
     &     LC(20),XJ(11),Z(NZ),LI(20),XZ(NZ),STRT(1),END(1),CHARL(1)
      DATA BLANK,DASH,PLUS,CAPI,ZERO / ' ','-','+','|','O' /
      DATA CHAR1,CHAR/'O','O','2','3','4','5','6','7','8','9','M'/
      DATA EPS / 1.E-10 /
      JINC = 5
      KINC = 6
      NR = 1 + 10 * JINC
      NC = 1 + 10 * KINC
      NRP1 = NR + 1
      NRM1 = NR - 1
      NCM1 = NC - 1
      FC = FLOAT(NCM1)
      NUNIT = IUNIT
      CHAR(1) = XCHAR
      IF ( CHAR(1) .EQ. BLANK ) CHAR(1) = CHAR1
      IF(IORD-1) 2,3,4
  2   CALL MIN(Y,NY,VMIN,IER)
      CALL MAX(Y,NY,VMAX,IER)
      IF(IZ .EQ. O ) GO TO 1
      CALL MIN(Z,NZ,ZMIN,IER)
      CALL MAX(Z,NZ,ZMAX,IER)
      IF(ZMIN.LT.VMIN)VMIN=ZMIN
      IF(ZMAX.GT.VMAX)VMAX=ZMAX
      GOTO 1
  3   VMIN = Y(1)
      VMAX = Y(NY)
      IF(IZ .EQ. O) GOTO 1
      ZMIN = Z(1)
      ZMAX = Z(NZ)
      IF(ZMIN .LT. VMIN) VMIN = ZMIN
      IF(ZMAX .GT. VMAX) VMAX = ZMAX
      GOTO 1
  4   VMIN = VMN
      VMAX = VMX
  1   CONTINUE
      RANGE = VMAX - VMIN
      IF ( RANGE .LT. EPS ) GOTO 99
      VINC = FLOAT(NRM1) / RANGE
      DO 10 I= 2,NRM1
      PARRAY(I,1)=CAPI
      PARRAY(I,NC) = CAPI
      DO 5 J=2,NCM1
      PARRAY(I,J)=BLANK
  5   CONTINUE
 10   CONTINUE
      DO 20 J=1,NC
      PARRAY(1,J) = DASH
      PARRAY(NR,J)=DASH
 20   CONTINUE
 15   J=0
      S=0.0
      DO 25 I=1,NC,KINC
      J=J+1
      XJ(J) = S
      S = S + .1
      PARRAY(NR,I)=PLUS
 25   CONTINUE
      DO 30 I = 1,NR,JINC
      PARRAY(NRP1-I,1) = PLUS
 30   CONTINUE
      IF ( NLINE .GE. 1 ) CALL PLINE(NR,NC,NLINE,STRT,END,CHARL,
     &                                 PARRAY,VMIN,VMAX,VINC)
 60   CONTINUE
      DO 45 K=1,NY
      INDX=X(K)*FC + 1.5
      INDY=(Y(K)-VMIN)*VINC + 1.5
      IF(IZ.NE.1)GO TO 70
      INDZ=(Z(K)-VMIN)*VINC + 1.5
      PARRAY(NRP1-INDY,INDX)=CHAR(1)
      PARRAY(NRP1-INDZ,INDX)=ZCHAR
      IF(INDZ.EQ.INDY)PARRAY(NRP1-INDY,INDX)=CHAR(10)
```

```
      GO TO 45
   70 CONTINUE
      IF ( CHAR(1) .EQ. XCHAR ) GOTO 44
      DO 42 I=1,10
      IF(PARRAY(NRP1-INDY,INDX).EQ.CHAR(I))GO TO 43
   42 CONTINUE
   44 PARRAY(NRP1-INDY,INDX)=CHAR(1)
      GO TO 45
   43 IF(I.EQ.10)I=I-1
      PARRAY(NRP1-INDY,INDX)=CHAR(I+1)
   45 CONTINUE
      IF ( I2 .NE. 2 ) GOTO 47
      DO 47 I = 1,N2
      INDX2 = X2(I) * FC + 1.5
      INDZ  = (Z(I) - VMIN) * VINC + 1.5
      IF ( PARRAY(NRP1-INDZ,INDX2) .EQ. CHAR(1) ) GOTO 48
      PARRAY(NRP1-INDZ,INDX2) = ZCHAR
      GOTO 47
   48 PARRAY(NRP1-INDZ,INDX2) = CHAR(10)
   47 CONTINUE
      WRITE(NUNIT,100)LC,L1
      DYX = (FLOAT(JINC) / FLOAT(NRM1)) * RANGE
      YXALUE = YMAX + DYX
      DO 300 I=1,NR,JINC
      IF(PARRAY(NRP1-I,1).NE.PLUS) PARRAY(NRP1-I,1)=ZERO
  300 CONTINUE
      DO 55 I=1,NRM1
      IF((PARRAY(I,1).EQ.PLUS).OR.(PARRAY(I,1).EQ.ZERO)) GO TO 50
      WRITE(NUNIT,105)(PARRAY(I,J),J=1,NC)
      GO TO 55
   50 YXALUE = YXALUE - DYX
      WRITE(NUNIT,110)YXALUE,(PARRAY(I,J),J=1,NC)
   55 CONTINUE
      WRITE(NUNIT,110)VMIN,(PARRAY(NR,J),J=1,NC)
      WRITE(NUNIT,115)XJ
      WRITE(NUNIT,120)LX,LY
      GO TO 999
   99 CONTINUE
      WRITE(NUNIT,199)LC
      WRITE(NUNIT,200)
  200 FORMAT(' X VALUES')
      WRITE(NUNIT,205)(X(I),I=1,NY)
  205 FORMAT(1X,10F10.5)
      WRITE(NUNIT,210)
  210 FORMAT(' Y VALUES')
      WRITE(NUNIT,205)(Y(I),I=1,NY)
  199 FORMAT(' ERROR IN PPLOT FOR THE PLOT OF ',20A4)
  999 RETURN
  100 FORMAT(1H1,15X,20A4,/,16X,20A4,/)
  105 FORMAT(13X,61A1)
  110 FORMAT(1X,F10.3,2X,61A1)
  115 FORMAT(12X,11(F3.1,3X))
  120 FORMAT(15X,'ABSCISSA IS ',2A4,4X,' , ORDINATE IS ',2A4)
      END
      SUBROUTINE PPLOTG(W,X,Y,Z,N,ICAPT,NAMW,NAMX,NAMY,NAMZ,MM,SMIN,
     +SMAX)
C*****************************************************************
C
C     SUBROUTINE TO PLOT LOCAL QUANTILES OF PERIODOGRAM FROM
C     SUBROUTINE QLOCAL. QUANTILES ARE PRINTED, LOG(QUANTILES)
C     ARE PLOTTED
C
C     INPUT  N,W,X,Y,Z - W IS ORDERED ON INPUT AND X(1)=X(W(1)),ETC.
C            ICAPT - LITERAL CONSTANT FOR CAPTION OF PLOT
C            NAMW,NAMX,NAMY,NAMZ - 4 CHAR. LABELS FOR W,X,Y,Z
C            MM - WIDTH OF PLOT (<=101)
C            SMIN,SMAX - MIN OF (X,Y,Z) AND MAX OF (X,Y,Z)
C
C     SUBROUTINES CALLED: FTERP, MAX, MIN
C
C*****************************************************************
C
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION W(N),X(N),Y(N),Z(N),ICAPT(20)
      DIMENSION W1(46),X1(46),Y1(46),Z1(46),AL(101)
      DATA BLANK,DOT,STAR,SL,PLUS,SM/1H ,1H.,1H*,1H|,1H+,1HM/
      IOPTX=0
      IF(N.GT.23) GO TO 20
      WRITE(IUNIT,10) N
   10 FORMAT(10X,'SAMPLE SIZE OF ',I2,' IS TOO SMALL IN PPLOTG.')
      GO TO 999
   20 CONTINUE
      AMIN=ALOG(SMIN)
      AMAX=ALOG(SMAX)
C
C     CREATE W1 VECTOR OF EQUALLY SPACED W AND INTERPOLATE TO OBTAIN
C     CORRESPONDING X,Y, AND Z VALUES
C
      DEC=(W(N)-W(1))/45.0
      DO 30 I=1,46
   30 W1(I)=W(1)+FLOAT(I-1)*DEC
      CALL FTERP(W,X,W1,X1,N,46)
      CALL FTERP(W,Y,W1,Y1,N,46)
      CALL FTERP(W,Z,W1,Z1,N,46)
C
C     INITIALIZE AL
C
      ON=(MM-1)/2
      DO 40 J=1,MM
   40 AL(J)=DOT
      WRITE(IUNIT,45) ICAPT
   45 FORMAT(//,T10,20A4,/)
      WRITE(IUNIT,50) NAMW,NAMX,NAMY,NAMZ,(AL(J),J=1,MM)
   50 FORMAT(//,16X,A4,6X,A4,6X,A4,6X,A4/10X,40(1H-),2X,101A1)
      DO 60 J=1,MM
   60 AL(J)=BLANK
      AL(1)=SL
      AL(MM)=SL
C
      RY=1.2*(AMAX-AMIN)
      IF(RY.LT.1.E-20) IOPTX=1
C
C     PLOT
C
      DO 100 J=1,46
      IF(IOPTX.EQ.1) GO TO 70
      C1=2.*(ALOG(X1(J))-AMIN)/RY - 1.
      C2=2.*(ALOG(Y1(J))-AMIN)/RY - 1.
      C3=2.*(ALOG(Z1(J))-AMIN)/RY - 1.
      GO TO 80
   70 C1=0.
      C2=0.
      C3=0.
   80 KX=ON*(C1+1.)+2.5
      KY=ON*(C2+1.)+2.5
      KZ=ON*(C3+1.)+2.5
      AL(KX)=DOT
      AL(KY)=STAR
      AL(KZ)=DOT
      IF(KX.EQ.KY) AL(KX)=SM
      IF(KX.EQ.KZ) AL(KX)=SM
      IF(KY.EQ.KZ) AL(KY)=SM
```

```
      WRITE(IUNIT,90) W1(J),X1(J),Y1(J),Z1(J),(AL(I),I=1,MM)
   90 FORMAT(10X,F10.4,1X,F9.4,1X,F9.4,1X,F9.4,2X,101A1)
      AL(KX)=BLANK
      AL(KY)=BLANK
      AL(KZ)=BLANK
  100 CONTINUE
      DO 110 I=1,MM
  110 AL(I)=DOT
      AL(1)=PLUS
      AL(MM)=PLUS
      WRITE(IUNIT,120) (AL(I),I=1,MM)
  120 FORMAT(10X,40(1H-),2X,101A1)
      PMAX=RY+AMIN
      WRITE(IUNIT,130) AMIN,PMAX
  130 FORMAT(47X,F10.4,60X,F10.4)
  999 CONTINUE
      RETURN
      END
      SUBROUTINE PREPSP(SPEC,NFREQS,IOPT,DIV,WK,SPMIN,SPMAX)
C***************************************************************
C
C     SUBROUTINE TO DIVIDE SPEC BY DIV, TAKE LOG, AND CLIP
C     TO -6,6  ALSO MAX AND MIN OF RESULT IS FOUND.
C
C     INPUT :
C               NFREQS,SPEC(1),...,SPEC(NFREQS),DIV
C               IOPT : 1 MEANS FREQS ARE 0 TO 2PI
C                      0 MEANS FREQS ARE 0 TO PI
C
C     OUTPUT :
C               WK(1),...,WK(N1) WHERE N1 IS (NFREQS/2)+1 OR NFREQS
C               SPMIN,SPMAX
C
C**********************************************************
C
      DIMENSION SPEC(NFREQS),WK(NFREQS)
C
C     INITIALIZE
C
      SPMIN=6.
      SPMAX=-6.
      DD=ALOG(DIV)
      EMIN=DIV*EXP(-6.0)
      EMAX=DIV*EXP(6.0)
      N1=(NFREQS/2)+1
      IF(IOPT.EQ.0) N1=NFREQS
C
C     TAKE LOG AND CLIP .
C
      DO 30 I=1,N1
      IF(SPEC(I).GT.EMAX) GO TO 10
      IF(SPEC(I).LT.EMIN) GO TO 20
      WK(I)=ALOG(SPEC(I))-DD
      IF(WK(I).LT.SPMIN) SPMIN=WK(I)
      IF(WK(I).GT.SPMAX) SPMAX=WK(I)
      GO TO 30
   10 CONTINUE
      WK(I)=6.
      SPMAX=6
      GO TO 30
   20 CONTINUE
      WK(I)=-6.
      SPMIN=-6.
   30 CONTINUE
C
C
      RETURN
      END
      FUNCTION QFIND(Q,N,QUANT)
C**********************************************************
C      PURPOSE    TO FIND THE PERCENTILE VALUE OF Q
C                 AT QUANT.
C      INPUT:
C            Q - VECTOR OF SIZE N
C            N - NUMBER OF VALUES IN Y
C            QUANT - QUANTILE VALUE
C**********************************************************
      COMMON /PARM1/ XMIN
      DIMENSION Q(N)
      P = FLOAT(N+1) * QUANT
      I = INT(P)
      P = P-I
      IF(I .NE. 0) GOTO 1
      QFIND = (1.-P) * XMIN + P * Q(1)
      RETURN
    1 IF(I .LT. N) GOTO 2
      QFIND = Q(N)
      RETURN
    2 QFIND = (1.-P) * Q(I) + P * Q(I+1)
      RETURN
      END
      SUBROUTINE QLOCAL(X,U,N,K,NAME)
C***************************************************************
C
C     SUBROUTINE TO COMPUTE AND DISPLAY LOCAL QUANTILES
C
C     INPUT  X - DATA SET FOR WHICH LOCAL QUANTILES ARE DESIRED
C            U - ABSCISSA FOR ORDINATE X (X IS A FUNCTION OF U)
C            N - DIMENSION OF VECTORS X AND U
C            K - SIZE OF LOCAL DATA BATCHES TAKEN FROM DATA BATCH X
C            NAME - HEADING FOR PLOTS
C            IPLTV - 1 IF VERSETEC PLOTS DESIRED, 0 OTHERWISE
C
C     OUTPUT  PRINTER AND VERSETEC PLOTS OF LOCAL QUANTILES G25,G50,
C             AND G75 - GP(I), P=25,50,75, IS THE P-TH PERCENTILE
C             VALUE FOR THE DATA BATCH X(I), X(I+1), ..., X(I+K-1).
C             THE ABSCISSA VALUE UG(I) IS TAKEN TO BE U(I+K/2)
C             IF K IS ODD AND THE AVERAGE OF U(I+K/2-1) AND
C             U(I+K/2) IF K IS EVEN.
C
C     SUBPROGRAMS CALLED: QUICK,QFIND,ICODEA,MIN,MAX,PPLOTG
C
C***************************************************************
      COMMON /UNIT/ IUNIT,NSCRCH
      DIMENSION X(N),U(N),NAME(20)
      DIMENSION Y(100),G25(550),G50(550),G75(550),UG(550)
      DIMENSION LAB1(20),LABEL(20,3)
      DATA LAB1/'LOCA','L QU','ANTI','LE P','ERIO','DOGR','AM: ',
     *'K = ',12*'    '/
      DATA NAMU,NAMG25,NAMG50,NAMG75/'  U ',' G25',' G50',' G75'/
C
C     IS K ODD OR EVEN?
C
      IODD=MOD(K,2)
      KD2=K/2
      KST=N-K+1
C
C     COMPUTE LOCAL QUANTILES
C
      DO 20 I=1,KST
      KL=I-1
      DO 10 J=1,K
```

```
      XMIN=QN(1)
      DO 90 I=1,8
      EXIQ(I) = QFIND(QN,NOP1,EXU(I))
   90 CONTINUE
      WRITE(NUNIT,92) EXU,EXIQ
   92 FORMAT(//T10,'  U  ',3X,8(F8.5,1X),
     +        /T10,'IQ(U)',3X,8(F8.5,1X),///)
      WRITE(NUNIT,900)
      CALL DESTAT(Y,N,NAME,LAB,LQUANT,0,0,X25,X50,X75)
      KASE=INUL*2
      ICASE(1)=LICASE(KASE-1)
      ICASE(2)=LICASE(KASE)
C   COMPUTE RAW SPACINGS (WK1=LITTLE Q)
      CALL QTOFQ(QN,U,NOP1,WK1,SPCFAC,AVLX)
C
C   FOR PERIODOGRAM, CHECK FOR SINUSOIDAL BEHAVIOUR
C
      IF(IOPT.EQ.0) GO TO 98
      CALL MAX(WK1,NQ,QMAXJ,INDQ)
      QP=QN(INDQ)/T10+X50
      WRITE(NUNIT,903) QMAXJ,QP
      DO 94 I=INDQ,N
      QP=QN(I)/T10+X50
      UQP=(FLOAT(I)-0.5)/FLOAT(N)
      WRITE(NUNIT,904) I,UQP,QP
   94 CONTINUE
   98 CONTINUE
C   COMPUTE AND PLOT WEIGHTED SPACINGS FOR CASE ICASE
      AVLGFQ = 0.
      DO 100 I = 1,NQ
      WK2(I) = FQFNC(U(I+1),INUL)
      AVLGFQ = AVLGFQ + ALOG(WK2(I))
  100 CONTINUE
      AVLGFQ = AVLGFQ / FLOAT(NQ)
      CALL WSPACE(WXS,D,NOP1,WK1,WK2,U,SW,AVLWX)
      SWLG = ALOG(SW)
      WRITE(NUNIT,902) AVLX,AVLWX,AVLGFQ,SW,SWLG
C   PLOT CUMULATIVE WEIGHTED SPACINGS WITH D+ AND D-
      CALL KSD(D,U,NOP1,DM,UM,DP,UP)
      CALL FCODEA(DP,8H(F7.4)  ,LAB9(2))
      CALL FCODEA(UP,8H(F7.4)  ,LAB9(6))
      CALL FCODEA(DM,8H(F7.4)  ,LAB9(10))
      CALL FCODEA(UM,8H(F7.4)  ,LAB9(14))
      LAB10(8)=ICASE(1)
      LAB10(9 )=ICASE(2)
      CALL PPLOT(U,D,NOP1,BLK,0.,0.,1.0,BLK,2,0.,1.,1.0,.1.,ASTER,
     &           NAMU,NAMCWS,LAB10,LAB9)
      WRITE(NUNIT,200)
  200 FORMAT(//////)
  900 FORMAT(///,T50,'FULLY  NON-PARAMETRIC  ANALYSIS'/T50,31(1H*))
  901 FORMAT(//T35,'SUMMARY OF AR PARAMETRIC SELECT ANALYSIS'/
     +        T35,40(1H-)///)
  902 FORMAT(///,T20,'AV. LOG SPACINGS',T40,
     +   'AV  LOG W  SPACINGS',T63,'AV. LOG HYP. FQ',
     +   T85,'SIGMA ZERO',T103,'LOG SIGMA ZERO'//
     +   ,T20,5(5X,G15.8))
  903 FORMAT(/,10X,'MAXIMUM JUMP FOR QUANTILE FUNCTION = ',F12.4,
     +' CORRESPONDING TO Q(U) = ',F12.5,//,14X,'I      U',19X,'Q',
     +/,10X,31(1H-))
  904 FORMAT(10X,I5,1X,F5.3,2X,F14.5)
      RETURN
      END
      SUBROUTINE QUICK(N,T)
C*******************************************************************
C   QUICK SORT THIS ALGORITH IS ALSO REFERRED TO AS A PARTITIONED
C   EXCHANGE SORT EXPECTED RUNTIME IS PROPORTIONAL TO N*LOG2(N)
C   ALTHOUGH THE WORST CASE IS PROPORTIONAL TO N**2.
C   REFERENCE: DONALD E. KNUTH- THE ART OF COMPUTER PROGRAMMING VOL 3.
C   INPUT .
C          X,N : VECTOR TO BE SORTED OF LENGTH N
C   OUTPUT :
C          X : SORTED VECTOR
C   SUBROUTINES CALLED : NONE
C*******************************************************************
      REAL T(N),Y
      INTEGER IP,LV(16),IV(16),LP,IUP
      LV(1)=1
      IV(1)=N
      IP=1
   10 IF(IP.LT.1) GO TO 75
   15 IF((IV(IP)-LV(IP)).LT.1) GO TO 20
      GO TO 25
   20 IP=IP-1
      GO TO 10
   25 LP=LV(IP)-1
      IUP=IV(IP)
      Y=T(IUP)
   30 IF((IUP-LP).LT.2) GO TO 45
      LP=LP+1
      IF(T(LP).LE.Y) GO TO 30
      T(IUP)=T(LP)
   35 IF((IUP-LP).LT.2) GO TO 40
      IUP=IUP-1
      IF(T(IUP).GE.Y) GO TO 35
      T(LP)=T(IUP)
      GO TO 30
   40 IUP=IUP-1
   45 T(IUP)=Y
      IF((IUP-LV(IP)).LT.(IV(IP)-IUP)) GO TO 55
      GO TO 60
   55 LV(IP+1)=LV(IP)
      IV(IP+1)=IUP-1
      LV(IP)=IUP+1
      GO TO 70
   60 LV(IP+1)=IUP+1
      IV(IP+1)=IV(IP)
      IV(IP)=IUP-1
   70 IP=IP+1
      GO TO 15
   75 RETURN
      END
      SUBROUTINE RELMN(CAT,M,DEC,MIN1,MIN2)
C***************************************************************
C
C   SUBROUTINE TO PRINT SIGNIFICANT RELATIVE MINIMA (AND THEIR
C   INDICES) OF CAT(1),...,CAT(M). A SIGNIFICANT RELATIVE
C   MINIMUM IS DEFINED TO BE A RELATIVE MINIMUM WHICH IS AT
C   LEAST DEC (USUALLY 1/SAMPLE SIZE) LESS THAN THE PREVIOUS
C   VALUE.
C
C   INPUT :
C            M,CAT(1),...,CAT(M),DEC
C
C   OUTPUT :
C            MIN1 : INDEX OF OVERALL MINIMUM OF CAT
C            MIN2 : INDEX (OTHER THAN MIN1) OF SMALLEST
C                   SIGNIFICANT RELATIVE MINIMUM (IF THERE ARE
C                   ANY) OR INDEX OF 2ND SMALLEST CAT VALUE.
C
C   SUBROUTINES CALLED : MIN
C
C****************************************************************
```

```
      XMIN=ON(1)                                                        ARSPIO
      DO 90 I=1,8                                                       ARSPIO
      EXIO(I) = OFIND(ON,NOP1,EXU(I))                                   ARSPIO
   90 CONTINUE                                                          ARSPIO
      WRITE(NUNIT,92) EXU,EXIO                                          ARSPIO
   92 FORMAT(//T10,'   U   ',3X,8(F8.5,1X),                             ARSPIO
     +        /T10,'IO(U)',3X,8(F8.5,1X),////)                          ARSPIO
      WRITE(NUNIT,900)                                                  ARSPIO
      CALL DESTAT(Y,N,NAME,LAB,LOUART,0,0,X25,X50,X75)                  ARSPIO
      KASE=INUL*2                                                       ARSPIO
      ICASE(1)=LICASE(KASE-1)                                           ARSPIO
      ICASE(2)=LICASE(KASE)                                             ARSPIO
C   COMPUTE RAW SPACINGS (WK1=LITTLE O)                                 ARSPIO
      CALL OTOFO(ON,U,NOP1,WK1,SPCFAC,AVLX)                             ARSPIO
C                                                                       ARSPIO
C   FOR PERIODOGRAM, CHECK FOR SINUSOIDAL BEHAVIOUR                     ARSPIO
C                                                                       ARSPIO
      IF(IOPT.EQ.0) GO TO 98                                            ARSPIO
      CALL MAX(WK1,NO,OMAXJ,INDO)                                       ARSPIO
      OP=ON(INDO)/T10+X50                                               ARSPIO
      WRITE(NUNIT,903) OMAXJ,OP                                         ARSPIO
      DO 94 I=INDO,N                                                    ARSPIO
      OP=ON(I)/T10+X50                                                  ARSPIO
      UOP=(FLOAT(I)-0.5)/FLOAT(N)                                       ARSPIO
      WRITE(NUNIT,904) I,UOP,OP                                         ARSPIO
   94 CONTINUE                                                          ARSPIO
   98 CONTINUE                                                          ARSPIO
C   COMPUTE AND PLOT WEIGHTED SPACINGS FOR CASE ICASE                   ARSPIO
      AVLGFO = 0                                                        ARSPIO
      DO 100 I = 1,NO                                                   ARSPIO
      WK2(I) = FOFNC(U(I+1),INUL)                                       ARSPIO
      AVLGFO = AVLGFO + ALOG(WK2(I))                                    ARSPIO
  100 CONTINUE                                                          ARSPIO
      AVLGFO = AVLGFO / FLOAT(NO)                                       ARSPIO
      CALL WSPACE(WKS,D,NOP1,WK1,WK2,U,SW,AVLWX)                        ARSPIO
      SWLG = ALOG(SW)                                                   ARSPIO
      WRITE(NUNIT,902) AVLX,AVLWX,AVLGFO,SW,SWLG                        ARSPIO
C   PLOT CUMULATIVE WEIGHTED SPACINGS WITH D+ AND D-                    ARSPIO
      CALL KSD(D,U,NOP1,DM,UM,DP,UP)                                    ARSPIO
      CALL FCODEA(DP,8H(F7.4)  ,LAB9(2))                                ARSPIO
      CALL FCODEA(UP,8H(F7.4)  ,LAB9(6))                                ARSPIO
      CALL FCODEA(DM,8H(F7.4)  ,LAB9(10))                               ARSPIO
      CALL FCODEA(UM,8H(F7.4)  ,LAB9(14))                               ARSPIO
      LAB10(8)=ICASE(1)                                                 ARSPIO
      LAB10(9 )=ICASE(2)                                                ARSPIO
      CALL PPLOT(U,D,NOP1,BLK,0.,0.,1.0,BLK,2,0.,1.,1,0.,1.,ASTER,      ARSPIO
     &            NAMU,NAMCWS,LAB10,LAB9)                               ARSPIO
      WRITE(NUNIT,200)                                                  ARSPIO
  200 FORMAT(//////)                                                    ARSPIO
  900 FORMAT(///,T50,'FULLY  NON-PARAMETRIC  ANALYSIS'/T50,31(1H=))     ARSPIO
  901 FORMAT(//T35,'SUMMARY OF AR PARAMETRIC SELECT ANALYSIS'/          ARSPIO
     +        T35,40(1H-)///)                                           ARSPIO
  902 FORMAT(///,T20,'AV. LOG SPACINGS',T40,                            ARSPIO
     +   'AV  LOG W. SPACINGS',T63,'AV. LOG HYP. FO',                   ARSPIO
     +   T85,'SIGMA ZERO',T103,'LOG SIGMA ZERO'//                       ARSPIO
     +   ,T20,5(5X,G15.8))                                              ARSPIO
  903 FORMAT(/,10X,'MAXIMUM JUMP FOR QUANTILE FUNCTION = ',F12.4,       ARSPIO
     +' CORRESPONDING TO Q(U) = ',F12.5,//,14X,'I      U',19X,'Q',      ARSPIO
     +/,10X,31(1H-))                                                    ARSPIO
  904 FORMAT(10X,I5,1X,F5.3,2X,F18.5)                                   ARSPIO
      RETURN                                                            ARSPIO
      END                                                               ARSPIO
      SUBROUTINE QUICK(N,T)                                             ARSPIO
C*******************************************************************     ARSPIO
C   QUICK SORT.THIS ALGORITH IS ALSO REFERRED TO AS A PARTITIONED       ARSPIO
C   EXCHANGE SORT.EXPECTED RUNTIME IS PROPORTIONAL TO N*LOG2(N)         ARSPIO
C   ALTHOUGH THE WORST CASE IS PROPORTIONAL TO N**2.                    ARSPIO
C   REFERENCE: DONALD E. KNUTH- THE ART OF COMPUTER PROGRAMMING VOL 3.  ARSPIO
C   INPUT :                                                             ARSPIO
C          X,N : VECTOR TO BE SORTED OF LENGTH N                        ARSPIO
C   OUTPUT :                                                            ARSPIO
C          X : SORTED VECTOR                                            ARSPIO
C   SUBROUTINES CALLED : NONE                                           ARSPIO
C*******************************************************************     ARSPIO
      REAL T(N),Y                                                       ARSPIO
      INTEGER IP,LV(16),IV(16),LP,IUP                                   ARSPIO
      LV(1)=1                                                           ARSPIO
      IV(1)=N                                                           ARSPIO
      IP=1                                                              ARSPIO
   10 IF(IP.LT.1) GO TO 75                                              ARSPIO
   15 IF((IV(IP)-LV(IP)).LT.1) GO TO 20                                 ARSPIO
      GO TO 25                                                          ARSPIO
   20 IP=IP-1                                                           ARSPIO
      GO TO 10                                                          ARSPIO
   25 LP=LV(IP)-1                                                       ARSPIO
      IUP=IV(IP)                                                        ARSPIO
      Y=T(IUP)                                                          ARSPIO
   30 IF((IUP-LP).LT.2) GO TO 45                                        ARSPIO
      LP=LP+1                                                           ARSPIO
      IF(T(LP).LE.Y) GO TO 30                                           ARSPIO
      T(IUP)=T(LP)                                                      ARSPIO
   35 IF((IUP-LP).LT.2) GO TO 40                                        ARSPIO
      IUP=IUP-1                                                         ARSPIO
      IF(T(IUP).GE.Y) GO TO 35                                          ARSPIO
      T(LP)=T(IUP)                                                      ARSPIO
      GO TO 30                                                          ARSPIO
   40 IUP=IUP-1                                                         ARSPIO
   45 T(IUP)=Y                                                          ARSPIO
      IF((IUP-LV(IP)).LT.(IV(IP)-IUP)) GO TO 55                         ARSPIO
      GO TO 60                                                          ARSPIO
   55 LV(IP+1)=LV(IP)                                                   ARSPIO
      IV(IP+1)=IUP-1                                                    ARSPIO
      LV(IP)=IUP+1                                                      ARSPIO
      GO TO 70                                                          ARSPIO
   60 LV(IP+1)=IUP+1                                                    ARSPIO
      IV(IP+1)=IV(IP)                                                   ARSPIO
      IV(IP)=IUP-1                                                      ARSPIO
   70 IP=IP+1                                                           ARSPIO
      GO TO 15                                                          ARSPIO
   75 RETURN                                                            ARSPIO
      END                                                               ARSPIO
      SUBROUTINE RELMN(CAT,M,DEC,MIN1,MIN2)                             ARSPIO
C************************************************************           ARSPIO
C                                                                       ARSPIO
C   SUBROUTINE TO PRINT SIGNIFICANT RELATIVE MINIMA (AND THEIR          ARSPIO
C   INDICES) OF CAT(1),...,CAT(M). A SIGNIFICANT RELATIVE               ARSPIO
C   MINIMUM IS DEFINED TO BE A RELATIVE MINIMUM WHICH IS AT             ARSPIO
C   LEAST DEC (USUALLY 1/SAMPLE SIZE) LESS THAN THE PREVIOUS            ARSPIO
C   VALUE.                                                              ARSPIO
C                                                                       ARSPIO
C   INPUT :                                                             ARSPIO
C           M,CAT(1),...,CAT(M),DEC                                     ARSPIO
C                                                                       ARSPIO
C   OUTPUT :                                                            ARSPIO
C           MIN1 : INDEX OF OVERALL MINIMUM OF CAT                      ARSPIO
C           MIN2 : INDEX (OTHER THAN MIN1) OF SMALLEST                  ARSPIO
C                  SIGNIFICANT RELATIVE MINIMUM (IF THERE ARE           ARSPIO
C                  ANY) OR INDEX OF 2ND SMALLEST CAT VALUE.             ARSPIO
C                                                                       ARSPIO
C   SUBROUTINES CALLED : MIN                                            ARSPIO
C                                                                       ARSPIO
C**************************************************************         ARSPIO
```

```
C
      DIMENSION CAT(M),REL(20),MININ(20)
      DATA NOUT/6/
C
C   FIND FIRST TWO MINIMA :
C
      CALL MIN(CAT,M,CMIN1,MIN1)
      CAT(MIN1)=0.0
      CALL MIN(CAT,M,CMIN2,MIN2)
      CAT(MIN1)=CMIN1
C
C   FIND SIGNIFICANT RELATIVE MINIMA :
C
      IF(M.LE.2) GO TO 50
      L=0
      MM1=M-1
      DO 20 I=2,MM1
      CC=CAT(I)
      IF((CC.GE.CAT(I-1)-DEC).OR.(CC.GE.CAT(I+1))) GO TO 20
      IF(I.EQ.MIN1) GO TO 20
      L=L+1
      REL(L)=CC
      MININ(L)=I
      WRITE(NOUT,10) L,I,CC
   10 FORMAT(10X,6HMININ(,I2,4H) = ,I3,2X,7HVALUE :,F15.7)
   20 CONTINUE
C
C   FIND MIN2 :
C
      IF(L.EQ.0) GO TO 50
      CALL MIN(REL,L,CMIN2,MIN2)
      MIN2=MININ(MIN2)
C
C
   50 CONTINUE
      WRITE(NOUT,60) MIN1,CMIN1,MIN2,CMIN2
   60 FORMAT(/,10X,7HMIN1 = ,I3,2X,12HCAT(MIN1) = ,F15.7/10X,
     17HMIN2 = ,I3,2X,12HCAT(MIN2) = ,F15.7)
C
C
      RETURN
      END
      SUBROUTINE SELREG(A,NDIM,K,NAME,KTOP,L,KP,COF,IND,IER)
C*****************************************************************
C
C   SUBROUTINE TO PERFORM SUBSET REGRESSION
C
C   INPUT :
C           A : COVARIANCE MATRIX TO BE ANALYZED
C           NDIM : DIMENSION OF A IN CALLING PROGRAM
C           K : NUMBER OF INDEPENDENT VARIABLES IN FULL
C           MODEL
C           NAME : NAME(I)=VARIABLE NUMBER OF VARIABLE ON
C           ITH DIAGONAL OF A.
C           KTOP : INDICATOR (0 MEANS SELREG AUTOMATICALLY
C           CHOOSES BEST MODEL, KTOP = M FORCES BEST M
C           VARIABLES INTO MODEL)
C           KTOP = -1 MEANS TERMINATE AFTER FIRST DELETION
C           IF RES VAR .LT. 0.100  OR TERMINATE IF RES VAR
C            .LT.0.001
C   IF KTOP.GE.99, CHISQ VALUES FOR P=.90 ARE USED AND
C   KTOP-100 IS USED TO DETERMINE OTHER OPTIONS.
C   IF KTOP.GE.199, THEN CHISQ VALUES FOR P=.80 ARE USED,
C   AS IS THE VALUE OF KTOP-200
C           L : SAMPLE SIZE (IF L IS NEGATIVE, NO PRINTING
C           IS DONE)
C
C   OUTPUT :
C           KP : NUMBER OF PREDICTOR VARIABLES IN CHOSEN
C           MODEL
C           IND : VECTOR TELLING WHICH VARIABLES (ACCORDING
C           TO NAME) HAVE BEEN CHOSEN
C           COF : VECTOR OF COEFFICIENTS FOR CHOSEN MODEL
C           IER : ERROR INDICATOR (1 MEANS ALL DIAGONAL
C           ELEMENTS OF A.LT.1.E-5 AT A CYCLE, 0 IS
C           NORMAL RETURN)
C
C   SUBROUTINES CALLED : NONE
C
C*****************************************************************
C
      DIMENSION A(NDIM,NDIM),NAME(K),COF(K),IND(K),
     1CHISQ(62),CB(100),IND1(100)
      DATA NOUT/6/
C
      IOPTP=1
      LLOPT=L
      IF(L.GT.0) GO TO 50
      L=-L
      IOPTP=0
   50 CONTINUE
      KTOP1=KTOP
      IF(KTOP.GE.99) GO TO 2
      CALL CHIP(1,CHISQ)
      GO TO 4
    2 IF(KTOP.GE.199) GO TO 3
      CALL CHIP(2,CHISQ)
      KTOP=KTOP-100
      GO TO 4
    3 CALL CHIP(3,CHISQ)
      KTOP=KTOP-200
    4 CONTINUE
C
      DO 10 I=1,K
      CB(I)=0.
   10 IND1(I)=0
      IER=0
      IF(IOPTP.EQ.0) GO TO 61
      WRITE(NOUT,41) K,KTOP
   41 FORMAT(//,10X,'REGRESSION ESTIMATION STAGEWISE SUMMARY'/10X,
     1'NUMBER OF VARIABLES IN FULL MODEL = ',I3/10X,
     1'MAXIMUM NUMBER OF COEFFICIENTS = ',I3/)
      WRITE(NOUT,1)
    1 FORMAT(10X,75(1H-))
   61 CONTINUE
      N=K+1
   44 FORMAT(10X,8(E10.3,2X))
C
C
C
      F2=6.635
      R1=0
      TOL=1.E-5
      KC=0
  100 KP=0
C
C
C   STAGE 1   ANALYZE MATRIX A
C
C
      RTOT=0.
      VMIN=2**30
```

```
      VMAX=0.
      NMIN=0
      NMAX=0
C
C     KTOL COUNTS NUMBER OF DIAGONALS.GT.TOL
C
      KTOL=0
      DO 110 IKT=1,K
      V=A(IKT,N)*A(N,IKT)
C
C     CHECK IF VARIABLE IKT IS IN MODEL
C
      IF(V.LT.0.) GO TO 105
      CB(IKT)=0.
      IND1(IKT)=0
      GO TO 106
  105 CB(IKT)=A(IKT,N)
      IND1(IKT)=1
      KP=KP+1
C
C     CHECK DIAGONAL A(IKT,IKT) :
C
  106 IF(A(IKT,IKT).LT.TOL) GO TO 110
      KTOL=KTOL+1
      V=V/A(IKT,IKT)
      IF(V.LT.0.) GO TO 60
      RTOT=RTOT+V
      GO TO 80
C
C     FIND VMIN,NMIN :
C
   60 IF((ABS(V)-ABS(VMIN)).GT.0.) GO TO 110
      VMIN=V
      NMIN=IKT
      GO TO 110
C
C     FIND VMAX,NMAX :
C
   80 IF((V-VMAX).LE.0.) GO TO 110
      VMAX=V
      NMAX=IKT
  110 CONTINUE
C
C     ARE ALL DIAGONALS.LT.TOL ?
C
      IF(KTOL.EQ.0) GO TO 999
C
C     CALCULATE AND PRINT CRITERION :
C
      PHI=L-KP-1
      KT1=K-KP
      IF(KT1.EQ.0) GO TO 120
      F1=CHISQ(KT1)
      GO TO 125
  120 F1=1.E-10
  125 T1=(ABS(VMIN)*PHI)/(F2*A(N,N))
      T2=(RTOT*PHI)/(F1*A(N,N))
C
C
      IF(NMIN.EQ.0) GO TO 130
      KD1=NAME(NMIN)
      GO TO 140
  130 KD1=0
  140 IF(NMAX.EQ.0) GO TO 150
      KD2=NAME(NMAX)
      GO TO 160
  150 KD2=0
  160 IF(K1.EQ.0) GO TO 170
      KD3=NAME(K1)
      GO TO 180
  170 KD3=0
C
C     AKAIKE :
C
  180 PHI0=L+2*KP
      PHI1=PHI0+2.
      PHI2=PHI0-2
      PHI3=2.*FLOAT(KP)/FLOAT(L)
      T3=ALOG(A(N,N))+PHI3
      T4=(VMAX*PHI1)/(2.*A(N,N))
      T5=(VMIN*PHI2)/(2.*A(N,N))
C
C
C
C     PRINT COEFFICIENTS AND VARIABLES
C
      IF(IOPTP.EQ.0) GO TO 52
      WRITE(NOUT,190)
  190 FORMAT(10X,12HVAR IN MODEL,10X,5HCOEFF)
   52 CONTINUE
      IF(KP.EQ.0) GO TO 184
      I1=1
      DO 191 I=1,K
      IF(IND1(I).EQ.0) GO TO 191
      I2=NAME(I)
      IND(I1)=I2
      COF(I1)=CB(I)
      I1=I1+1
      IF(IOPTP.EQ.0) GO TO 191
      WRITE(NOUT,192) I2,CB(I)
  192 FORMAT(10X,I12,5X,F10.5)
  191 CONTINUE
  184 IF(IOPTP.EQ.0) GO TO 55
      WRITE(NOUT,185)
  185 FORMAT(10X,4X,8HCRIT ADD,6X,8HCRIT DEL,7X,7H    AIC,
     17X,7HFPE ADD,7X,7HFPE DEL)
      WRITE(NOUT,186) T2,T1,T3,T4,T5
  186 FORMAT(10X,5(F12.6,2X))
      WRITE(NOUT,187)
  187 FORMAT(1X,9X,3X,7HVAR ADD,5X,7HVAR DEL,4X,8HVAR LAST,3X,
     19HNO. PREDS,5X,7HRES VAR,5X,7HNO. CYC)
      WRITE(NOUT,188) KD2,KD1,KD3,KP,A(N,N),KC
  188 FORMAT(8X,4I12,2X,F10.5,2X,I10)
      WRITE(NOUT,1)
   55 CONTINUE
C
C
C     STAGE 2 : CHOSE PIVOT OR TERMINATE
C
C
  200 IF((KTOP.GT.0).AND.(KP.LT.KTOP).AND.(NMAX.EQ.0)) GO TO 919
      IF((KTOP.GT.0).AND.(KP.LT.KTOP).AND.(NMAX.NE.0)) GO TO 300
      IF((KTOP.GT.0).AND.(KP.EQ.KTOP)) GO TO 280
C
C     AUTOMATIC :
C
C     CHECK FOR DELETE :
C
C
C     CHECK KTOP=-1 CONDITIONS :
C
      IF(KTOP.NE.-1) GO TO 224
```

```
      IF((KP.LT.KC).AND.(A(N,N).LT.0.1)) GO TO 280
      IF(A(N,N).LT.0.001) GO TO 225
C
C   CHECK DELETE CONDITIONS FOR KTOP.NE.-1 :
C
 224  IF((ABS(VMIN).GE.F2*A(N,N)/PHI).OR.(K1.EQ.NMIN)) GO TO 230
C
C   YES
C
 225  K1=NMIN
      GO TO 240
C
C   NO
C
C
C   CHECK ADD :
C
 230  IF(RTOT.LE.F1*(A(N,N)/PHI)) GO TO 280
C
C   YES
C
 300  K1=NMAX
 240  CONTINUE
C
C
C   STAGE 3 : PIVOT
C
C
      PIVOT=1./A(K1,K1)
      DO 350 JKT=1,N
 350  A(K1,JKT)=A(K1,JKT)*PIVOT
      DO 370 IKT=1,N
      IF(IKT.EQ.K1) GO TO 370
      TEMP=A(IKT,K1)
         DO 360 JKT=1,N
 360     A(IKT,JKT)=A(IKT,JKT)-TEMP*A(K1,JKT)
      A(IKT,K1)=-TEMP*PIVOT
 370  CONTINUE
      A(K1,K1)=PIVOT
C
C   RETURN  TO STAGE 1
C
      KC=KC+1
      IF(KC.LT.100) GO TO 100
C
C   TERMINATIONS :
C
      WRITE(NOUT,913)
 913  FORMAT(/,10X,'TERMINATION OF SELREG SINCE KC.GE.100')
      GO TO 280
C
C
  919 WRITE(NOUT,920)
 920  FORMAT(/,10X,'TERMINATION SINCE ALL REMAINING VARIABLES HAVE 0 PAR
     1TIAL VARIANCE')
      GO TO 280
C
C
  999 WRITE(NOUT,1000)
 1000 FORMAT(/,10X,'TERMINATION SINCE ALL DIAGONALS.LT.TOL')
      IER=1
 280  KTOP=KTOP1
      L=LLOPT
      RETURN
C
C
      END
      SUBROUTINE SLMX(KP,IND,COF,ALPHA,BETA,NORDAR,NORDMA)
C****************************************************************
C
C   SUBROUTINE TO TRANSFORM MIXED SELECT SCHEME PARAMETERS
C   KP,IND,COF TO FULL MIXED SCHEME PARAMETERS NORDAR,NORDMA,
C   ALPHA,BETA
C
C   INPUT :
C             KP : NUMBER OF LAGS IN SELECT SCHEME
C             IND(1),...,IND(KP) : LAGS (AR LAGS +, MA LAGS -)
C             COF(1),...,COF(KP) : COEFFS CORRESPONDING TO LAGS
C
C   OUTPUT :
C             NORDAR,NORDMA : AR AND MA PART ORDERS
C             ALPHA(1),...,ALPHA(NORDAR) : AR COEFFS
C             BETA(1),...,BETA(NORDMA) : MA COEFFS
C
C****************************************************************
C
      DIMENSION IND(KP),COF(KP),ALPHA(1),BETA(1)
C
C   FIND NORDAR,NORDMA :
C
      NORDAR=0
      NORDMA=0
      IF(KP.EQ.0) GO TO 99
      DO 20 I=1,KP
      II=IND(I)
      IF(II.GT.0) GO TO 10
      IF(-II.GT.NORDMA) NORDMA=-II
      GO TO 20
   10 CONTINUE
      IF(II.GT.NORDAR) NORDAR=II
   20 CONTINUE
C
C   ZERO OUT ALPHA AND BETA :
C
      IF(NORDAR.EQ.0) GO TO 40
      DO 30 I=1,NORDAR
      ALPHA(I)=0.0
   30 CONTINUE
   40 CONTINUE
      IF(NORDMA.EQ.0) GO TO 60
      DO 50 I=1,NORDMA
      BETA(I)=0.0
   50 CONTINUE
   60 CONTINUE
C
C   PUT COEFFS INTO ALPHA,BETA :
C
      DO 80 I=1,KP
      II=IND(I)
      IF(II.GT.0) GO TO 70
      IJ=-II
      BETA(IJ)=COF(I)
      GO TO 80
   70 CONTINUE
      ALPHA(II)=COF(I)
   80 CONTINUE
C
C   FINISH :
C
   99 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SPCV(SPEC,NFREQS,M,CT,ST,R,R0)
C*****************************************************************
C
C     SUBROUTINE TO CALCULATE THE AUTOCOVARIANCES R0,R(1),...,R(M)
C     FROM THE CORRESPONDING SPECTRAL DENSITY (SPEC)
C     EVALUATED AT THE NFREQS EQUALLY SPACED FREQUENCIES
C     BETWEEN 0 AND TWOPI.
C
C     INPUT :
C             NFREQS (INTEGER WHOSE LARGEST PRIME FACTOR IS
C             < 23), M, SPEC(1),...,SPEC(NFREQS)
C
C     OUTPUT :
C             R0,R(1),...,R(M)
C
C     AUXILLIARY
C             CT,ST
C
C     SUBROUTINES CALLED   FFT
C
C*****************************************************************
C
      DIMENSION SPEC(NFREQS),CT(NFREQS),ST(NFREQS),R(M)
C
      FAC=(8.*ATAN(1.0))/FLOAT(NFREQS)
C
      DO 1 I=1,NFREQS
      CT(I)=SPEC(I)
    1 ST(I)=0.
C
      CALL FFT(CT,ST,NFREQS,NFREQS,NFREQS,-1)
C
      R0=FAC*CT(1)
      DO 2 I=1,M
    2 R(I)=FAC*CT(I+1)
C
      RETURN
      END
      SUBROUTINE SPPLT(SPEC,NFREQS,FFREQ,DELF,IOPT)
C*******************************************************************
C
C     SUBROUTINE TO PRINTER PLOT THE SPECTRAL QUANTITIES
C     SPEC(1),...,SPEC(NFREQS) WHERE THE FREQUENCIES ARE
C     GIVEN BY FFREQ+(J-1)*DELF, J=1,NFREQS.
C
C     INPUT :
C             NFREQS,FFREQ,DELF
C             SPEC(1),...,SPEC(NFREQS)
C             IOPT : 1 MEANS LOG SPEC PLOTTED, SPEC PRINTED
C
C     SUBROUTINES CALLED : MAX,MIN
C
C*******************************************************************
C
      DIMENSION SPEC(NFREQS),AL(61)
      DATA NOUT,BLANK,DOT,X/6,1H ,1H.,1H*/
C
C     SCALE :
C
      IF(SPEC(1).LT.1.E-5) SPEC(1)=SPEC(2)
      IF(IOPT.NE.1) GO TO 5
      SMAX=ALOG(SPEC(1))
      SMIN=SMAX
      DO 1 I=2,NFREQS
      C=ALOG(SPEC(I))
      IF(C.GT.SMAX) SMAX=C
    1 IF(C.LT.SMIN) SMIN=C
      SRANGE=SMAX-SMIN
      GO TO 7
    5 CALL MAX(SPEC,NFREQS,SMAX,IND)
      CALL MIN(SPEC,NFREQS,SMIN,IND)
      SRANGE=SMAX-SMIN
    7 CONTINUE
C
C     INCREMENT FOR 2 PAGES :
C
      NINT=1
      IF(NFREQS.GT.120) NINT=(NFREQS/120)+1
C
C     INITIALIZE AND PRINT HEADING :
C
      WRITE(NOUT,10)
   10 FORMAT(/,10X,1X,9HFREQUENCY,6X,6HPERIOD,8X,
     14HSPEC/10X,34(1H-),4X,61(1H.))
      DO 20 I=1,61
   20 AL(I)=BLANK
C
C     PLOT :
C
      IF(SRANGE.GT.1.E-25) GO TO 8
      WRITE(NOUT,9)
    9 FORMAT(/,10X,'RANGE.LT.1.E-25 IN SPPLT')
      GO TO 99
    8 CONTINUE
      TWOPI=8.*ATAN(1.0)
      DO 30 I=1,NFREQS,NINT
      FRQ=(FFREQ+FLOAT(I-1)*DELF)/TWOPI
      PER=0.
      IF(FRQ.GT.1.E-10) PER=1./FRQ
      SS=SPEC(I)
      IF(IOPT.EQ.1) SS=ALOG(SS)
      C=2.*(((SS-SMIN)/SRANGE)-.5)
      J=30.*(C+1.)+1.5
      AL(J)=X
      WRITE(NOUT,25) FRQ,PER,SPEC(I),AL
   25 FORMAT(10X,F10.5,2X,F10.5,2X,F10.5,4X,61A1)
      AL(J)=BLANK
   30 CONTINUE
C
C
   99 CONTINUE
      RETURN
      END
      SUBROUTINE WSPACE(WXS,CWXS,NOP1,XS,FOOO,U,SIGO,AVLWX)
C*****************************************************************
C     SUBROUTINE TO COMPUTE D(U), CUMULATIVE D'S, AND SIGMAO
C     FOR THE MODEL Q(U)=MU+SIGMA*QO(U)
C     INPUT :
C             FQ,NQ : VECTOR OF LENGTH NQ CONTAINING FQ(U)
C             FOOO : HYPOTHESIZED DENSITY QUANTILE FROM FOFNC
C             U : VECTOR OF LENGTH NQ CONTAINING U VALUES
C     OUTPUT :
C             WXS : VECTOR OF LENGTH NQ CONTAINING D(U)
C             CWXS : VECTOR OF LENGTH NQ CONTAINING THE
C                    CUMULATIVE D'S
C             SIGO : COMPUTED VALUE OF SIGMAO = CWXS(NQ)
C             AVLWX : AVERAGE LOG OF LITTLE D(U)
C     SUBROUTINES CALLED : NONE
C*****************************************************************
      DIMENSION FOOO(NOP1),XS(NOP1),U(NOP1),WXS(NOP1),CWXS(NOP1)
      NQ=NOP1-1
      CWXS(1)=0.
```

```
      DO 10 I= 1,NO                                          ARSP10
      WKS(I) = FOOO(I) * RS(I)                               ARSP10
      CWKS(I+1) = CWKS(I) + WKS(I)                           ARSP10
   10 CONTINUE                                               ARSP10
      S = 0.                                                 ARSP10
      FN = FLOAT(NO)                                         ARSP10
      D1 = 1./CWKS(NOP1)                                     ARSP10
      D2 = D1 * FN                                           ARSP10
      DO 20 I=1,NO                                           ARSP10
      WKS(I)=WKS(I) * D2                                     ARSP10
      CWKS(I)=CWKS(I) * D1                                   ARSP10
      S = S + ALOG(WKS(I))                                   ARSP10
   20 CONTINUE                                               ARSP10
      CWKS(NOP1)=1                                           ARSP10
      AVLWK = S/FN                                           ARSP10
      SIGO = 1./D2                                           ARSP10
      RETURN                                                 ARSP10
      END                                                    ARSP10
```

REFERENCES

Box, G. E. P. and Jenkins, G. M. (1970) Time Series Analysis, Forecasting and Control. Holden Day: San Francisco.

Newton, H. Joseph (1983) An Introduction to the Methods of Time Series Analysis in the Time and Frequency Domains: A User's Guide to the TIMESBOARD Computing Library. Institute of Statistics, Texas A&M University.

Parzen, Emanuel (1981) "Time Series Model Identification and Prediction Variance Horizon," Proceedings of Second Tulsa Symposium on Applied Time Series Analysis. Academic Press: New York, p. 415-447.

Parzen, Emanuel (1983) "Time Series Model Identification by Estimating Information, Memory, and Quantiles," Technical Report No. A-22, Department of Statistics, Texas A&M University.

Parzen, Emanuel, and Anderson, Scott (1980) "ONESAM, A Computer Program for Nonparametric Data Analysis and Density Quantile Estimation," Technical Report No. B-1, Institute of Statistics, Texas A&M University.